

NASA CR-158907

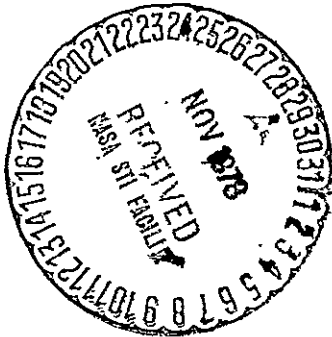
SONIC-BOX METHOD EMPLOYING LOCAL MACH NUMBER  
FOR OSCILLATING WINGS WITH THICKNESS

(NASA-CR-158907) SONIC-BOX METHOD EMPLOYING  
LOCAL MACH NUMBER FOR OSCILLATING WINGS WITH  
THICKNESS Final Report (Lockheed-Georgia  
Co., Marietta.) 73 p HC A04/MF A01 CSCL 01A

N79-10999

Unclass  
G3/02 36971

By S. Y. Ruo



September 1978

Prepared under Contract No. NAS1-13613 by

LOCKHEED-GEORGIA COMPANY  
Marietta, Georgia

for Langley Research Center

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION

## CONTENTS

	<u>Page</u>
SUMMARY . . . . .	1
INTRODUCTION . . . . .	1
SYMBOLS . . . . .	3
METHOD . . . . .	4
COMMENTS ON THE PROGRAM . . . . .	6
RESULTS . . . . .	8
Delta Wing . . . . .	8
Rectangular Wing . . . . .	9
CONCLUDING REMARKS . . . . .	10
REFERENCES . . . . .	12
APPENDIX: COMPUTER PROGRAM . . . . .	24
Input Guide . . . . .	25
Sample Input . . . . .	29
Sample Output . . . . .	31
Program Listing . . . . .	34

# SONIC-BOX METHOD EMPLOYING LOCAL MACH NUMBER FOR OSCILLATING WINGS WITH THICKNESS

By S. Y. Ruo  
Lockheed-Georgia Company

## SUMMARY

A computer program has been developed to account approximately for the effects of finite wing thickness in the transonic potential flow over an oscillating wing of finite span. The program is based on the original sonic-box program of Rodemich and Andrew and has been extended to include the effects of the swept trailing edge and the thickness of the wing. Account for the non-uniform flow caused by finite thickness is made by application of the local linearization concept. The thickness effect, expressed in terms of the local Mach number, is included in the basic solution to replace the coordinate transformation method used in the earlier work. Calculations were made for a delta wing and a rectangular wing performing plunge and pitch oscillations, and the results were compared with those obtained from other methods. An input guide and a complete listing of the computer code are presented.

## INTRODUCTION

In reference 1, the sonic-box method computer program was developed for calculation of unsteady transonic flow aerodynamics for oscillating planar wings with unswept trailing edge by approximating the wing planform with a matrix of square boxes. Later, it was extended to include the swept trailing edge and control surfaces in reference 2. The sonic-box method uses a doublet velocity potential as the basic solution to satisfy the linearized transonic flow, unsteady small-perturbation velocity-potential equation with the associated boundary conditions.

In references 3 and 4, the wing thickness effect is partially recovered by the inclusion of local Mach number in the governing equation for the

unsteady transonic flow. It uses the concept of local linearization to reduce the nonlinear small-perturbation equation to a linear one with non-constant coefficients. This is further reduced to a linear equation with constant coefficients by an appropriate coordinate transformation. This final equation and the associated boundary condition in the transformed space become identical to those treated in the physical space by Rodemich and Andrew in reference 1. The numerical results for the wing with thickness were obtained by adopting the sonic-box method in the transformed space. Because of the assumptions made in deducing the governing equation to a manageable form, this technique is applicable only to relatively thin wings. That is, the local mean Mach number on the wing surface must not be very different from unity. Further, it is assumed that there is no flow separation and no strong shock waves on the wing surface.

The computer programs developed in references 1, 2, and 3 use the least-square method to fit some of the input data, such as wing deflection or steady Mach number distribution on the wing, and to fit the computed velocity potential with a form of pre-determined polynomial surface for the subsequent calculation of the unsteady pressure and the generalized aerodynamic force coefficients. The computer program described in reference 4 adopts the natural cubic spline for fitting calculated velocity potential and the spline-surface for fitting input modal deflections and Mach number distribution instead of the polynomial-surface fitting used in references 1, 2, and 3. The codes in references 3 and 4 allow the computation of generalized aerodynamic force coefficients for wings of zero and finite thickness; the swept trailing edges are allowed but not the control surfaces.

The computer program presented in this report is developed according to the "alternate technique" described in reference 5. The coordinate transformation technique as used in references 3 and 4 fails when the mean local Mach number on the wing becomes very different from unity. In order to avoid this problem, an alternate technique was proposed in reference 5 to approximately account for the thickness effect, expressed in terms of mean local Mach number on the wing, by including it directly in the basic doublet solution to replace the coordinate transformation. The computer program thus developed is smaller

than that of reference 4 and the amount of computation required has also been reduced.

In general, the basic assumptions and limitations applied to the computer code in reference 4 also apply to the present code. However, the present formulation avoids the difficulties associated with the artificial wake and wing-surface fold-over due to multivalued transformation which limits the usefulness of the coordinate transformation formulation of reference 4. The zero thickness wing portion of the computer code is unchanged from that of reference 4. Input is identical in both codes and the output differs very little between them.

#### SYMBOLS

$b$	reference length (dimension = L)
$C_p$	pressure coefficient
$\exp, e$	exponential function
$i$	$\sqrt{-1}$
$k$	reduced frequency, $\omega b/U_\infty$
$L$	unit of length
$L_{ij}$	generalized aerodynamic force coefficient
$M$	local Mach number
$T$	unit of time
$U_\infty$	reference velocity (freestream), (dimension = L/T)
$x, y, z$	dimensionless Cartesian coordinates (reference length = $b$ )
$\theta_{ij}$	phase angle of $L_{ij}$
$\tau$	maximum thickness to root chord ratio
$\phi_0$	magnitude of oscillatory dimensionless small perturbation velocity potential
$\omega$	angular velocity (dimension = radian/T)
$( )_{le}$	subscripts denote quantity at leading edge
$( )_{te}$	subscripts denote quantity at trailing edge

## METHOD

The computer program described in reference 4 is based on the coordinate transformation technique to reduce the locally linearized equation with non-constant coefficients for nonzero thickness wing at sonic speed to a linear one with constant coefficients. This linear equation with the associated boundary conditions can, then, be solved with sonic-box method. When the mean local Mach number on the wing becomes very different from that of the free-stream, the transformation may become multivalued and consequently an artificial wake or wing-surface fold-over may be created in the transformed space. This technique fails once it happens. In order to avoid this problem, an alternate technique was proposed in reference 5 to approximately account for the thickness effect, expressed in terms of mean local Mach number on the wing, by including it in the basic doublet solution.

The governing equation for unsteady transonic small perturbation velocity potential is

$$\varphi_{Oyy} + \varphi_{Ozz} - M^2(2ik\varphi_{Ox} - k^2\varphi_O) = 0, \quad (1)$$

where

$$\varphi_O(x, y, z) = \varphi(x, y, z, t) \cdot e^{-ikt},$$

which is also equation (1) of reference 4.

The basic solution for equation (1), representing a point doublet oriented parallel to the z-axis at the origin and satisfying the required condition at infinity for a small finite region on the wing where the value of M, the Mach number, is considered to be constant, may be written as

$$\varphi_O = \begin{cases} 0, & x \leq 0, \\ \frac{ik}{2\pi} \frac{zM^2}{x^2} \exp \left\{ -\frac{1}{2}ik \left[ x + \frac{M^2(y^2+z^2)}{x} \right] \right\}, & x > 0, \end{cases} \quad (2)$$

in which  $M$  is regarded as a parameter. This solution satisfies equation (1) only in a small finite region of the wing; so the solution may be considered to be of the locally linearized form.

The only quantity in the program of reference 4 requiring modification is the velocity influence coefficient for the wings with thickness. It is presently written as

$$A = \frac{ik}{2\pi} M^2 \iint_E \frac{1}{(x-\xi)^2} \exp \left\{ -\frac{ik}{2} \left[ (x-\xi) + \frac{M^2(y-\eta)^2}{(x-\xi)} \right] \right\} d\xi d\eta \quad (3)$$

$$= \frac{ik}{2\pi} M \iint \frac{1}{u^2} \exp \left[ -\frac{ik}{2} \left( u + \frac{v^2}{u} \right) \right] dudv \quad (4)$$

where

- $H$  = length of the box side
- $k$  = reduced frequency
- $M$  = mean local Mach number
- $i = \sqrt{-1}$
- $l = kH$
- $E$  = box at  $(\xi, \eta)$
- $u = (x-\xi)/H$
- $v = M(y-\eta)/H$

The value of the velocity influence coefficient computed in the sonic-box computer program is with  $M = 1.0$  in equations (3) and (4). Under this condition, the velocity influence coefficient is function of the wing geometry only. For  $M \neq 1.0$ , it becomes function of the Mach number also. The value of the modified velocity influence coefficient required in this alternate technique to account for the wing thickness effect may be evaluated from the table computed for  $M = 1.0$  condition for the same reduced frequency.

To evaluate the modified velocity influence coefficient for this alternate technique, one may do the following:

1. take the average value of the mean local Mach number at the center of the receiving,  $(x, y)$ , and sending,  $(\xi, \eta)$ , boxes,

2. multiply the spanwise distance between these two box centers by the value of the average Mach number,
3. interpolate the modified velocity influence coefficient from the original table for  $M = 1.0$  with the value of the modified spanwise distance,  $v$ ,
4. multiply this value by the mean local Mach number at the center of the sending box.

The rest of the computation remains practically unchanged except that the computation in the transformed space is totally eliminated.

#### COMMENTS ON THE PROGRAM

The velocity potential influence coefficients for a wing of zero thickness at a given frequency are only a function of the geometry. However, in addition to the geometry, they are also a function of the local Mach number distribution for the nonzero thickness wing under present formulation. It may be possible to perform the integration in equation (3) analytically with a new formula or with that already in the earlier program with some approximation. No attempt was made to derive the totally new formulation. One of the approximate methods which was studied but not implemented in the present program is to substitute the local Mach number,  $M$ , in the integrand of equation (3) with  $(1-\epsilon)$ , where  $\epsilon$  is a positive or a negative small number. After expanding the exponential function involving  $\epsilon$  term and neglecting all  $\epsilon^2$  or higher terms, one obtains an approximate form of the integrand, for a doublet at the origin and  $z = 0$ , as follows:

$$\left(\frac{1}{x^2} + i\epsilon k \frac{y^2}{x^3}\right) \cdot \exp \left[-\frac{ik}{2} \left(x + \frac{y^2}{x}\right)\right]. \quad (5)$$

The exponential function in equation (5) is the same as that used in the case for  $M = 1.0$  and the routines in the earlier sonic-box computer program may be utilized to perform the integration. Due to its complexity, and the additional computer storage and time required, this approximate method was not adopted to generate the new velocity influence coefficient matrix with the Mach number effect. Instead, it is interpolated from the velocity influence



coefficient matrix for the zero thickness wing as described in the preceding section. The Mach number appearing in the exponential function in equation (3) is only associated with the distance between receiving and sending points and it is regarded as to modify the effective distance between these two points. Therefore, the average Mach number is used to maintain its interchangeability. Another Mach number in equation (3) is regarded as to modify the doublet strength. Since the integration is performed over the surface of the sending box, it is logical to use the Mach number at that point. This simplification in coupling the Mach number effect enables a reduction of the size of the computer code and the computation time. The computed results appeared to be reasonable under the assumptions of small perturbation theory and local linearization concepts.

In the present formulation, it implies as in the coordinate transformation formulation (ref. 4) that the Mach number variation in the spanwise direction is not large. The accuracy of these methods decreases when a large Mach number variation in spanwise direction exists. The present method, however, does not fail abruptly as does the coordinate transformation method when spanwise variation of Mach number becomes large enough to cause multi-valued transformation and hence fold-over of wing-surface.

Since no smoothing has been applied on either the input data or any computed values in the data fitting process during the computation, the calculated unsteady pressure coefficient distribution may not be smooth and should be used with caution. In order to use it, the computed pressure coefficient should be put through a smoothing process such as the smoothing portion of the two-dimensional cubic-spline fitting routines in the present program. The pressure coefficient is obtained by differentiation of a set of numerical values whereas the generalized aerodynamic force coefficient is obtained by integration. Since integration itself is a smoothing process, the resulting generalized aerodynamic force coefficient is considered to be acceptable within the bounds of the accuracy of the numerical techniques and the adequacy of the sonic-box method. The option of data smoothing is not provided in the three-dimensional spline-surface fitting process used in the present code for

input data such as wing deflections and mean local Mach number. The spline-surface is required to go through all input points.

## RESULTS

Sample calculations are made for a delta wing and a rectangular wing oscillating in plunge (Mode 1) and in pitch about the apex (Mode 2). The mean angle of attack is zero and the freestream is at sonic speed.

### Delta Wing

The delta wing considered here is a flattened elliptic cone of aspect ratio 1.5 and thickness-to-root-chord ratio  $\tau = 0.1$ . Convergence with respect to the number of boxes along the root chord for the generalized force coefficients ( $L_{ij}$ ) due to plunge and pitch about the apex, at a reduced frequency of  $k = 0.2$ , is shown in figure 2. The maximum numerical difference within the applied range of 15 and 30 boxes along the root chord is about 4 percent, and the trend of convergence with and without thickness is essentially the same. Based on the results shown in figure 2, it appears that the gain in convergence by using a large number of boxes to represent the wing is not obvious as compared with a fortuitous selection of the number of boxes to use. The numerical fluctuation in the convergence plot is largely caused by the box arrangement along the wing leading edge which, in turn, is dependent on the number of boxes selected for use along the wing root chord. Contribution from the partial boxes along the leading edge has been taken into account, but the fluctuation still exists.

The variation of each force coefficient, using 30 boxes along the root chord, versus the reduced frequency is plotted in figure 3. The results from figure 7 of reference 4 are also shown. The numerical difference between the results for wings with and without thickness is very small, generally less than one percent. This is a result of the Mach number, at each box-center used in the computation, lying within the narrow range of 0.92 and 0.98 in chordwise direction and remaining constant in spanwise direction (see fig. 8

of ref. 4). However, the thickness effect on flutter speed can be significant (ref. 6). The results for the case with thickness obtained from the present method and that of reference 4 are not very different.

### Rectangular Wing

The rectangular wing considered here has aspect ratio 2.0 and a biconvex (circular arc) airfoil with thickness-to-chord ratio  $\tau = 0.0521$ . The variation of each force coefficient, using 20 boxes along the root chord, versus the reduced frequency is plotted in figure 4. The results obtained from the present method, and from references 4 and 7, are included in the figure. The thickness effect on the rectangular wing is seen to be slightly larger than that on the delta wing. This is probably caused by the wide range of Mach number variation (fig. 6) on the rectangular wing, even though the thickness ratio is only 0.0521 for the rectangular wing against 0.1 for the delta wing.

The present method predicts values higher than either Landahl's results (ref. 7) for the zero thickness case or the results of reference 4 for the non-zero thickness case. The difference of the generalized aerodynamic force coefficients for the nonzero thickness case between the results obtained from the present method and that of reference 4 is quite large. This might be caused by the difference in interpretation of the effective distance between the sending and the receiving points in the present method and the coordinate transformation method used in reference 4. It is felt, however, that the interpretation used in the present program is more physically sound than that used in reference 4. The phase angle predicted by the sonic-box method at very low reduced frequency becomes meaningless when the magnitude of any force coefficient approaches to zero with decreasing reduced frequency (for example, see figs. 4(b) and 4(d)). This is due to numerical inaccuracy and not to any inadequacy of the method.

The steady-state pressure coefficient obtained from reference 8 for the rectangular wing considered here is shown in figure 5(a) for the chordwise (x-direction) distribution and in figure 5(b) for the spanwise (y-direction) distribution. The corresponding Mach numbers at the box-centers, interpolated

from the fitted spline-surface, are plotted in figures 6(a) and 6(b). The interpolated values deviate from the input data more near the leading edge than near the trailing edge. This probably was caused by the use of more dense spacing of input points near the leading edge as compared with those near the trailing edge in chordwise direction and by the lack of input points near the leading edge in spanwise direction, especially in the in-board portion of the wing. A better fit than that shown in figures 6(a) and 6(b) may be obtained by using more evenly spaced input points than those shown in figures 5(a) and 5(b).

#### CONCLUDING REMARKS

A sonic-box method computer program is presented for the application of a local linearization concept capable of accounting approximately for wing thickness effects in unsteady sonic flow. The thickness effect, expressed in terms of the local steady Mach number, is directly included in the basic solution. The local doublet strength is adjusted from the sonic flow condition to that for the local flow, and the governing equation is reduced to the one used in the original sonic-box method for zero thickness wings. Thus, the original sonic-box method concept can be used directly to treat nonzero thickness wings.

Convergence of the numerical results with respect to the number of boxes used in representing the wing planform seemed to depend more on the arrangement of the boxes along a swept leading edge than on the total number of boxes used, even though the partial boxes along the leading edge were included in the computation. For a wing with unswept leading edge, the use of a small number of boxes (say, 15 to 20 along the root chord) appeared to be sufficient to obtain results that were essentially converged.

When the input data require spline-surface fitting, the input points must be selected in such a way that they are as uniformly spaced as possible to avoid locally-concentrated large errors. A smoothing option for the two-dimensional cubic-spline has been included in the present program, but it was

not utilized in the sample runs shown in this report. Since the box method itself is numerical in nature, the distribution of calculated values may not always be smooth; and it may become necessary to perform the smoothing before any gradients are evaluated.

Based on the sample runs made, the contribution due to thickness was not found to be very large in comparison with the results calculated from the coordinate transformation method. Due to the lack of reliable experimental data, it is rather difficult to assess the validity of the present approach in accounting for thickness effects.

## REFERENCES

1. Rodemich, E. R. and Andrew, L. V.: Unsteady Aerodynamics for Advanced Configurations, Part II - A Transonic Box Method for Planar Lifting Surfaces. FDL-TDR-64-152, Part II, May 1965, Air Force Flight Dynamics Lab., Wright-Patterson Air Force Base, Ohio.
2. Stenton, T. E. and Andrew, L. V.: Transonic Unsteady Aerodynamics for Planar Wings with Trailing Edge Control Surfaces. AFFDL-TR-67-180, August 1968, Air Force Flight Dynamics Lab., Wright-Patterson Air Force Base, Ohio.
3. Ruo, S. Y.: Calculation of Unsteady Transonic Aerodynamics for Oscillating Wings with Thickness. (Computer Program). NASA CR-132477, Sept. 1974.
4. Ruo, S. Y.: Improved Sonic-Box Computer Program for Calculating Transonic Aerodynamic Loads on Oscillating Wings with Thickness. NASA CR-158906, Sept. 1978.
5. Ruo, S. Y. and Theisen, J. G.: Calculation of Unsteady Transonic Aerodynamics for Oscillating Wings with Thickness. NASA CR-2259, June 1975.
6. Ruo, S. Y.; Yates, E. C., Jr.; and Theisen, J. G.: Calculation of Unsteady Transonic Aerodynamics for Oscillating Wings with Thickness. *AIAA Journal of Aircraft*, Vol. 11, No. 10, Oct. 1974, pp. 601-608.
7. Landahl, M. T.: *Unsteady Transonic Flow*. Pergamon Press, New York, 1961.
8. Alksne, A. Y. and Spreiter, J. R.: Theoretical Pressure Distributions on Wings of Finite Span at Zero Incidence for Mach Number Near 1. NASA TR R-88, 1960.

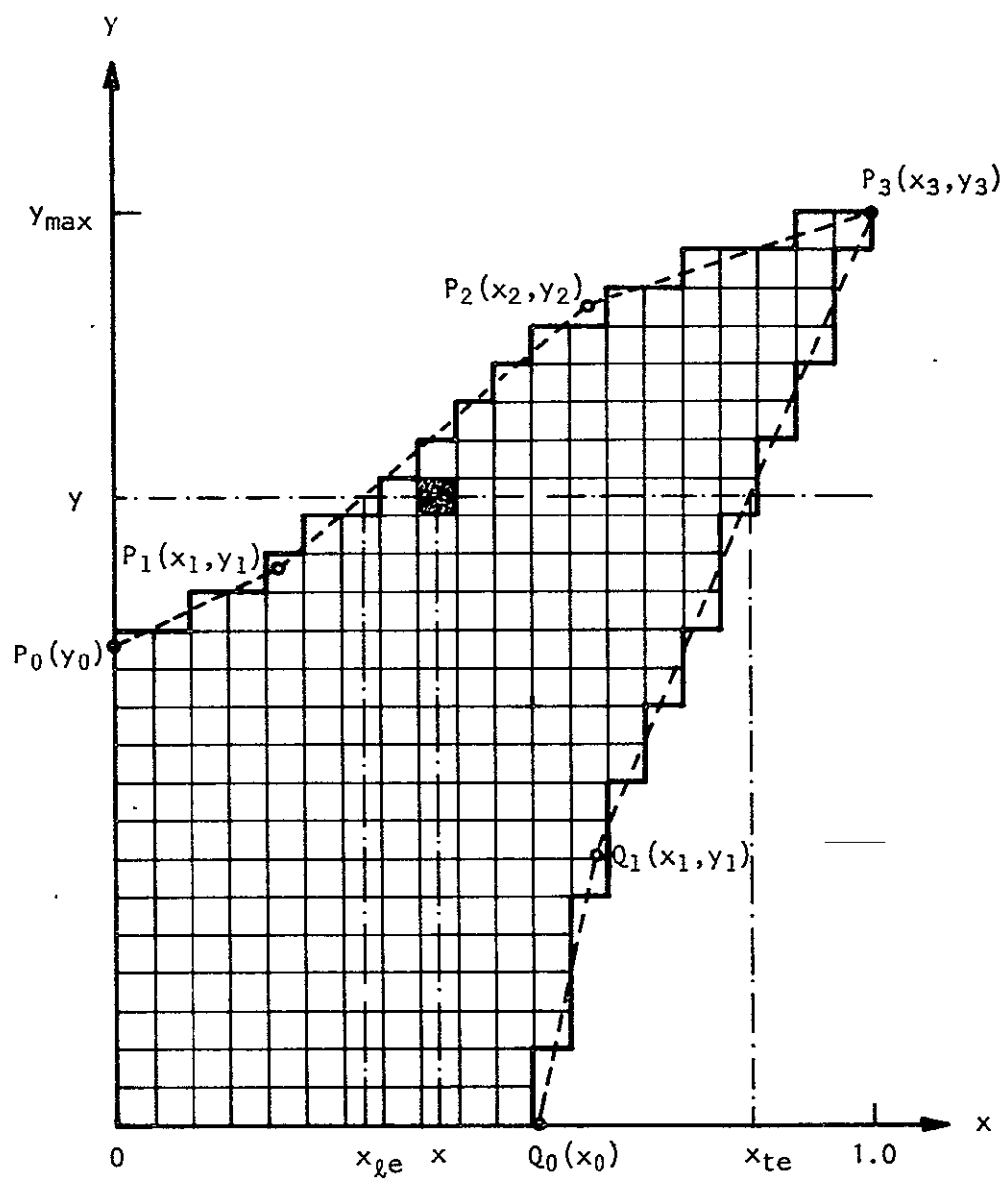


Figure 1. - Half wing geometry.

ORIGINAL PAGE IS  
OF POOR QUALITY

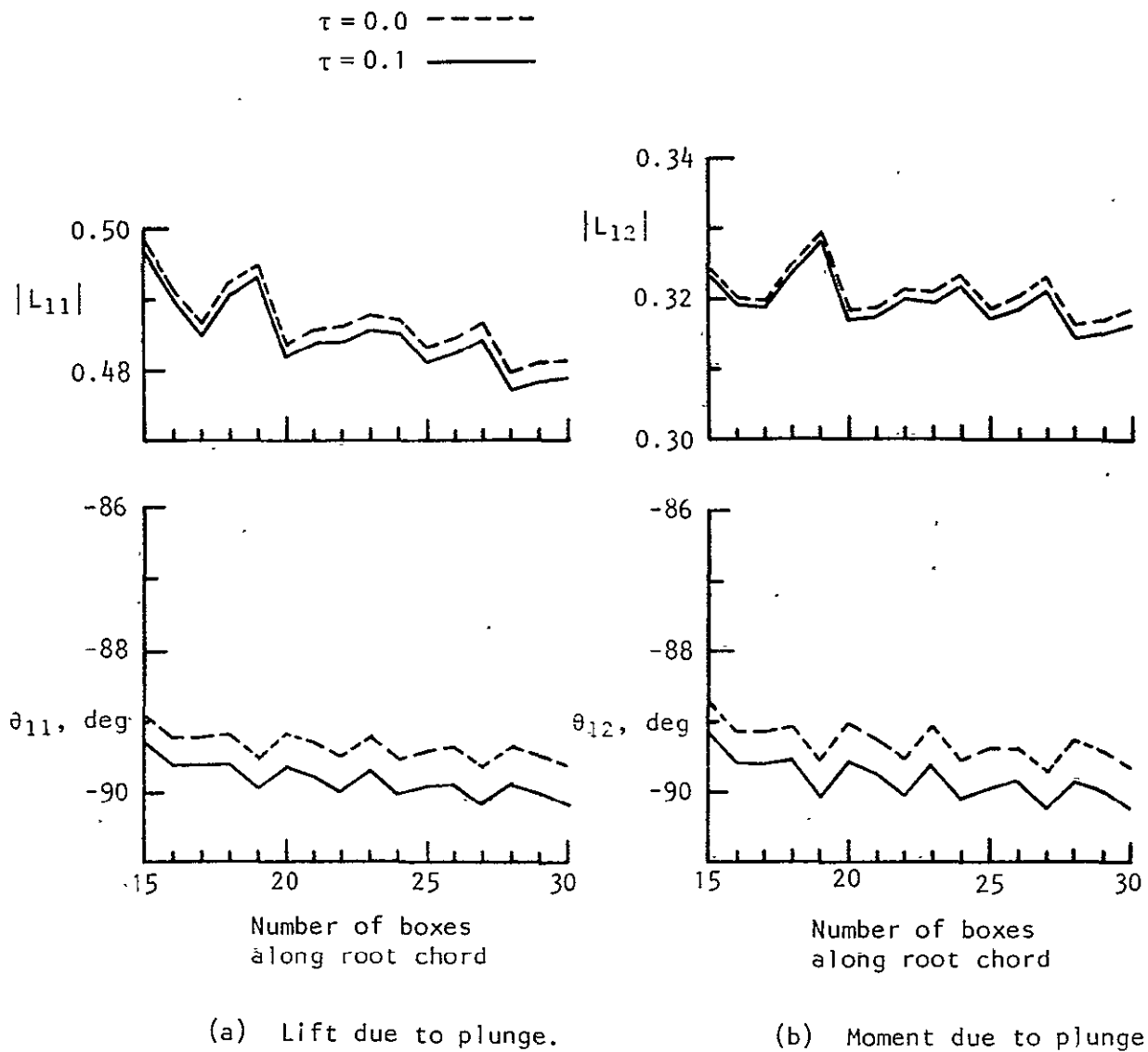
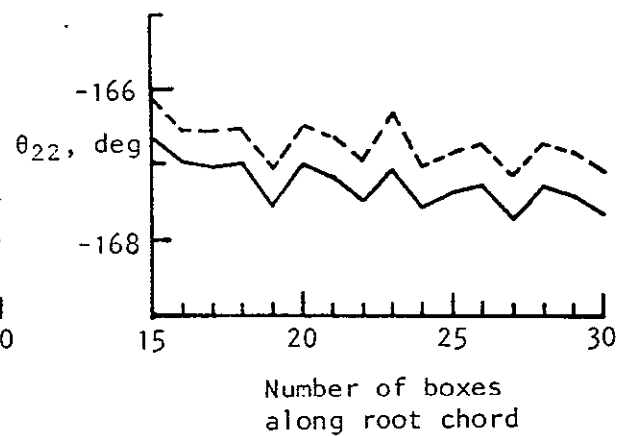
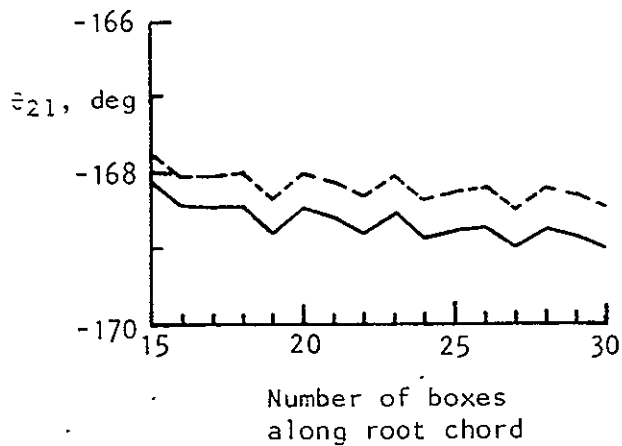
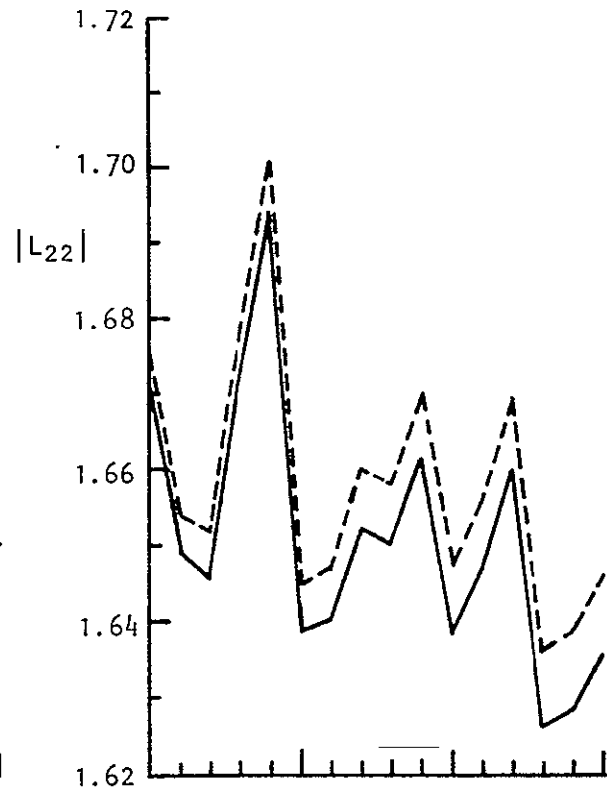
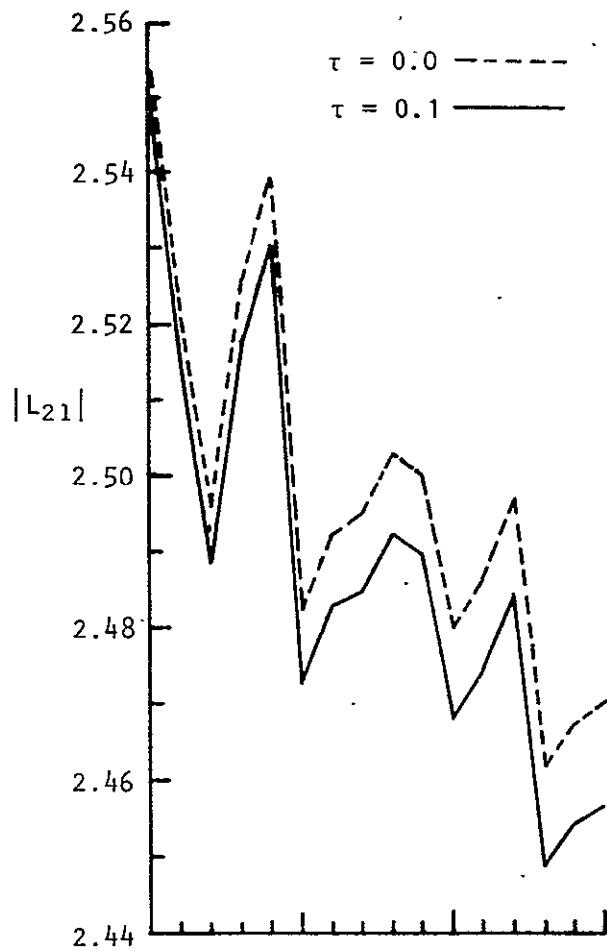


Figure 2. - Convergence of force coefficients due to plunge and pitch for delta wing of aspect ratio 1.5 at reduced frequency 0.2.





(c) Lift due to pitch.

(d) Moment due to pitch.

Figure 2. - Concluded.

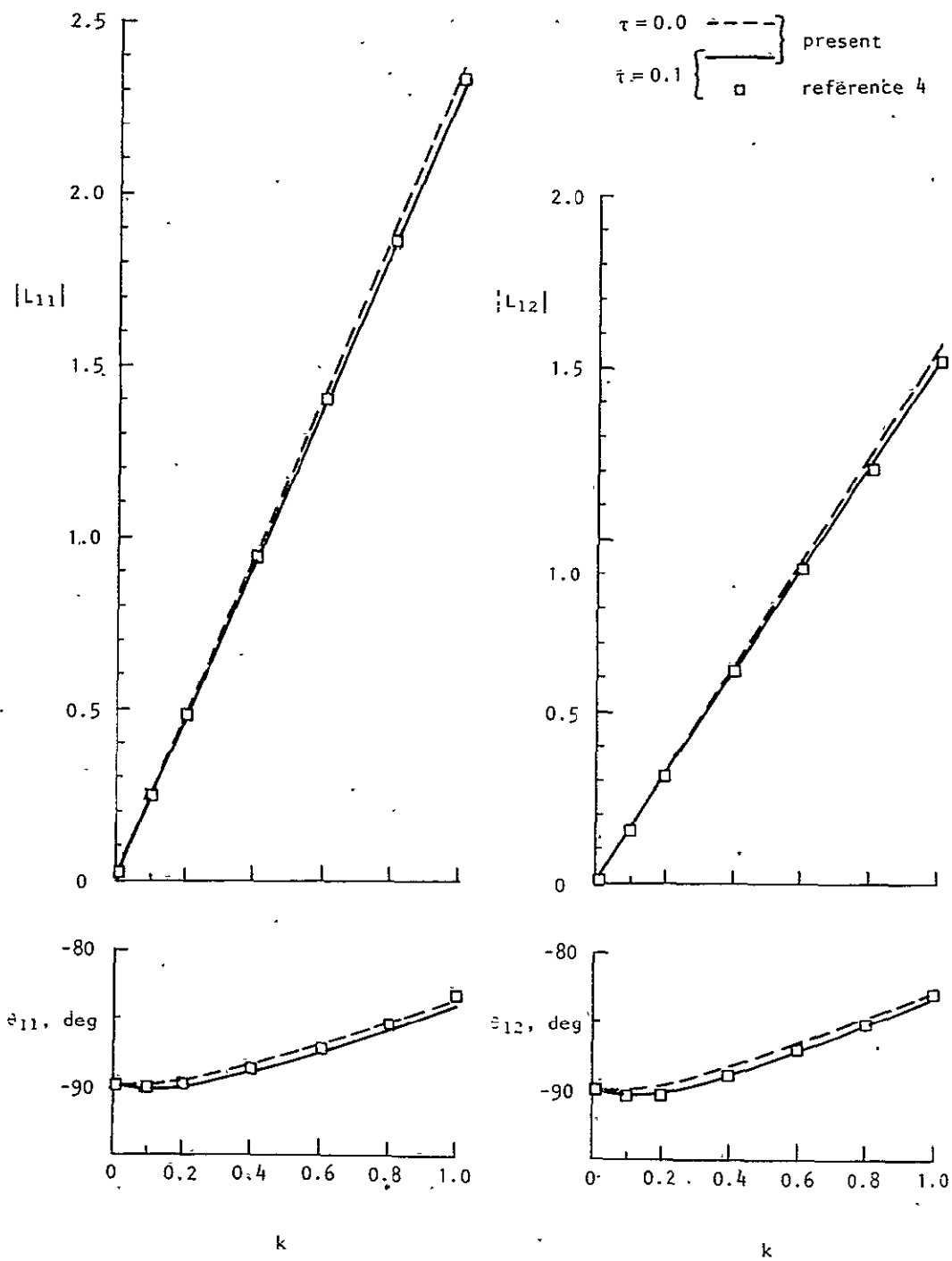
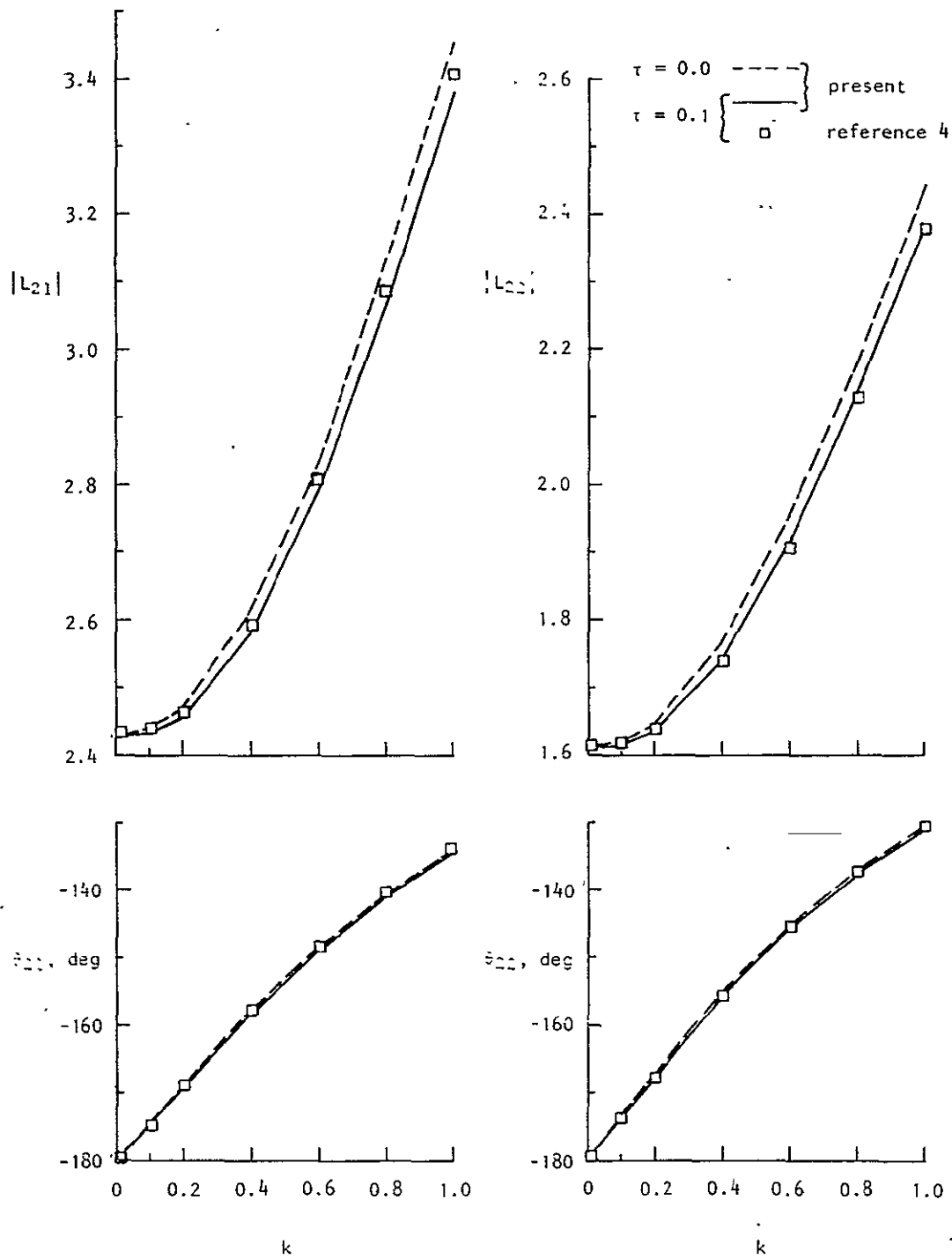


Figure 3. - Force coefficients due to plunge and pitch for delta wing of aspect ratio 1.5 with 30 boxes along root chord.



(c) Lift due to pitch.

(d) Moment due to pitch.

Figure 3. - Concluded.

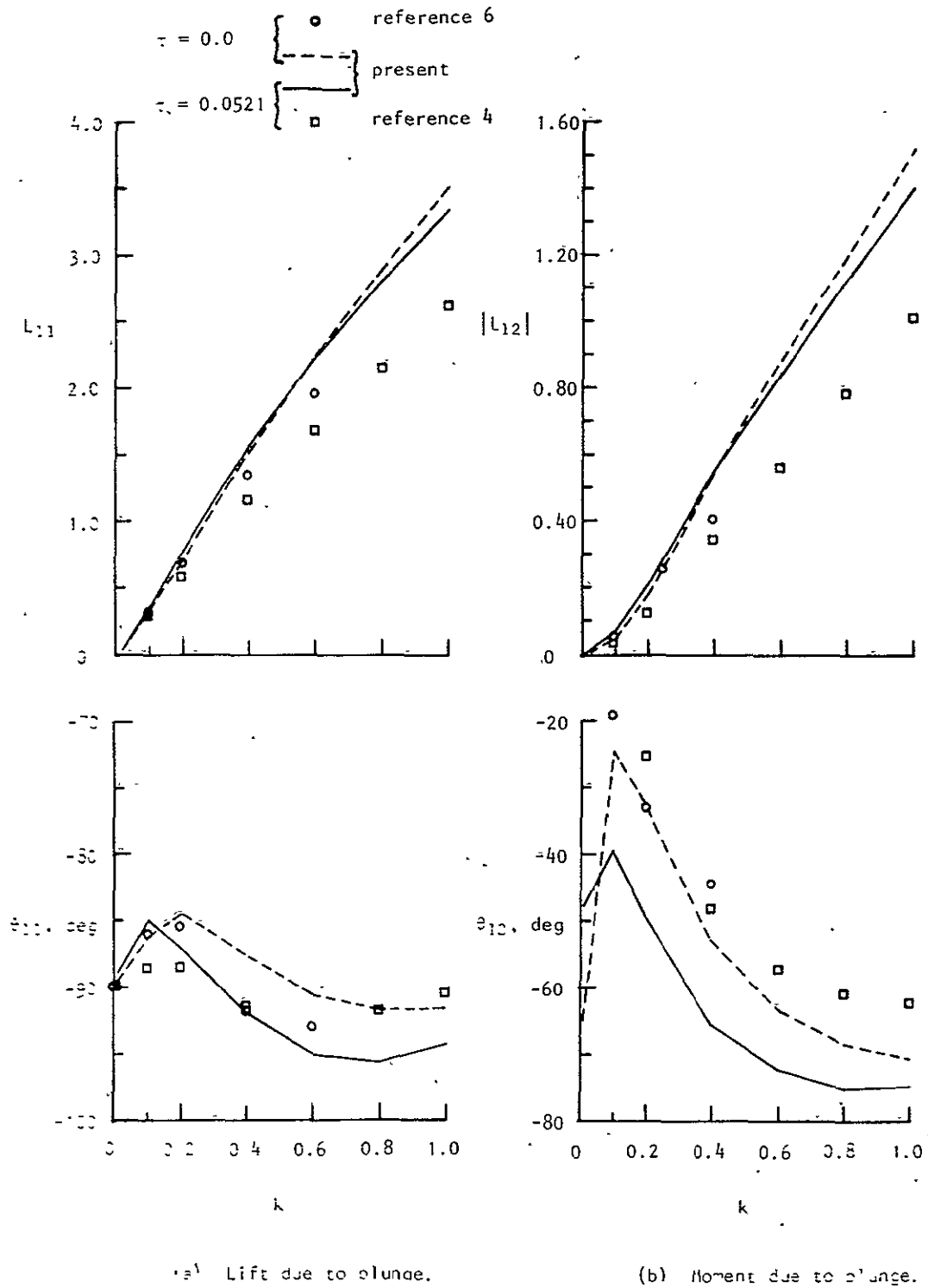


Figure 4. - Force coefficients due to plunge and pitch for rectangular wing of aspect ratio 2.0 with 20 boxes along root chord.

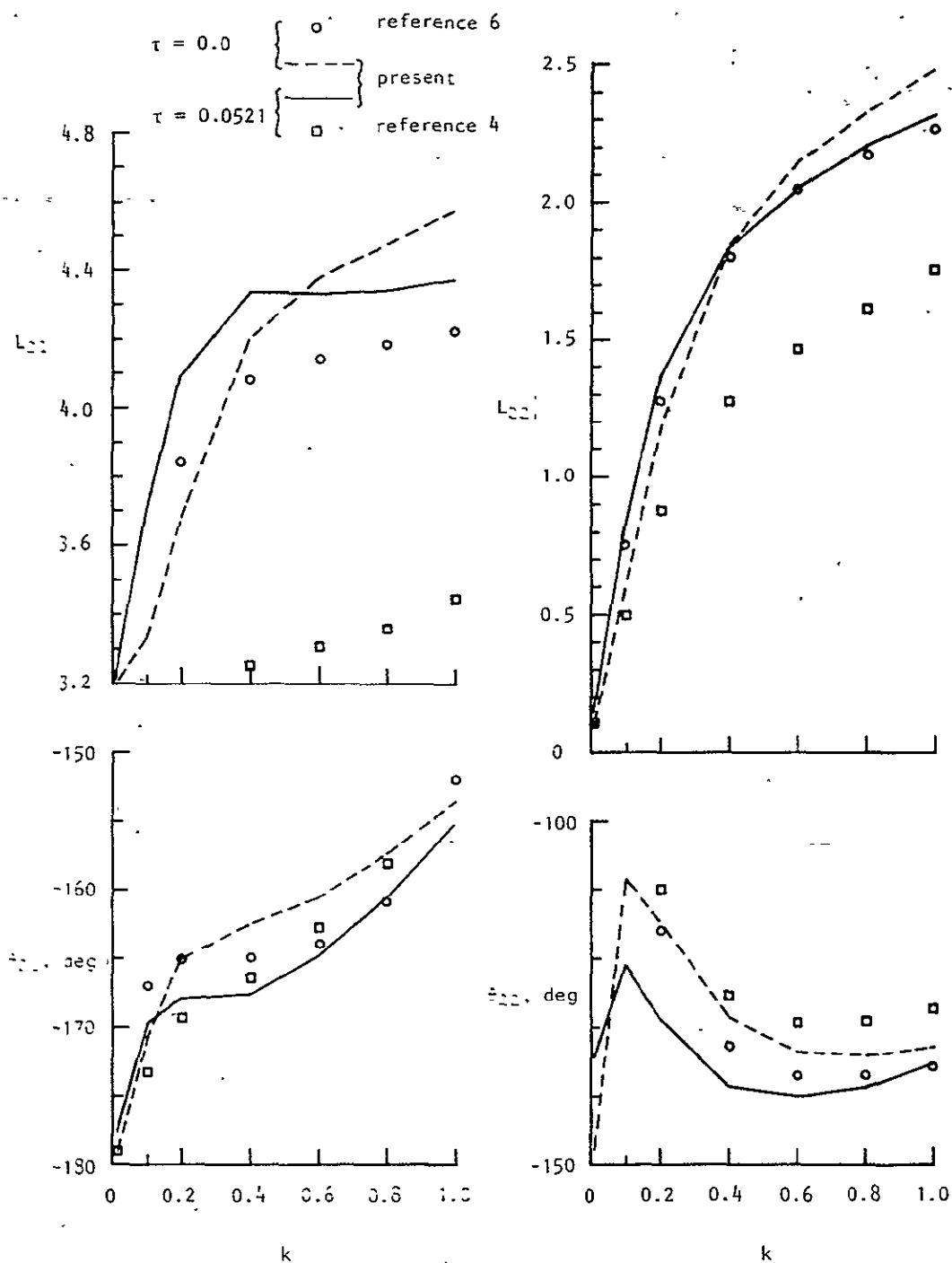
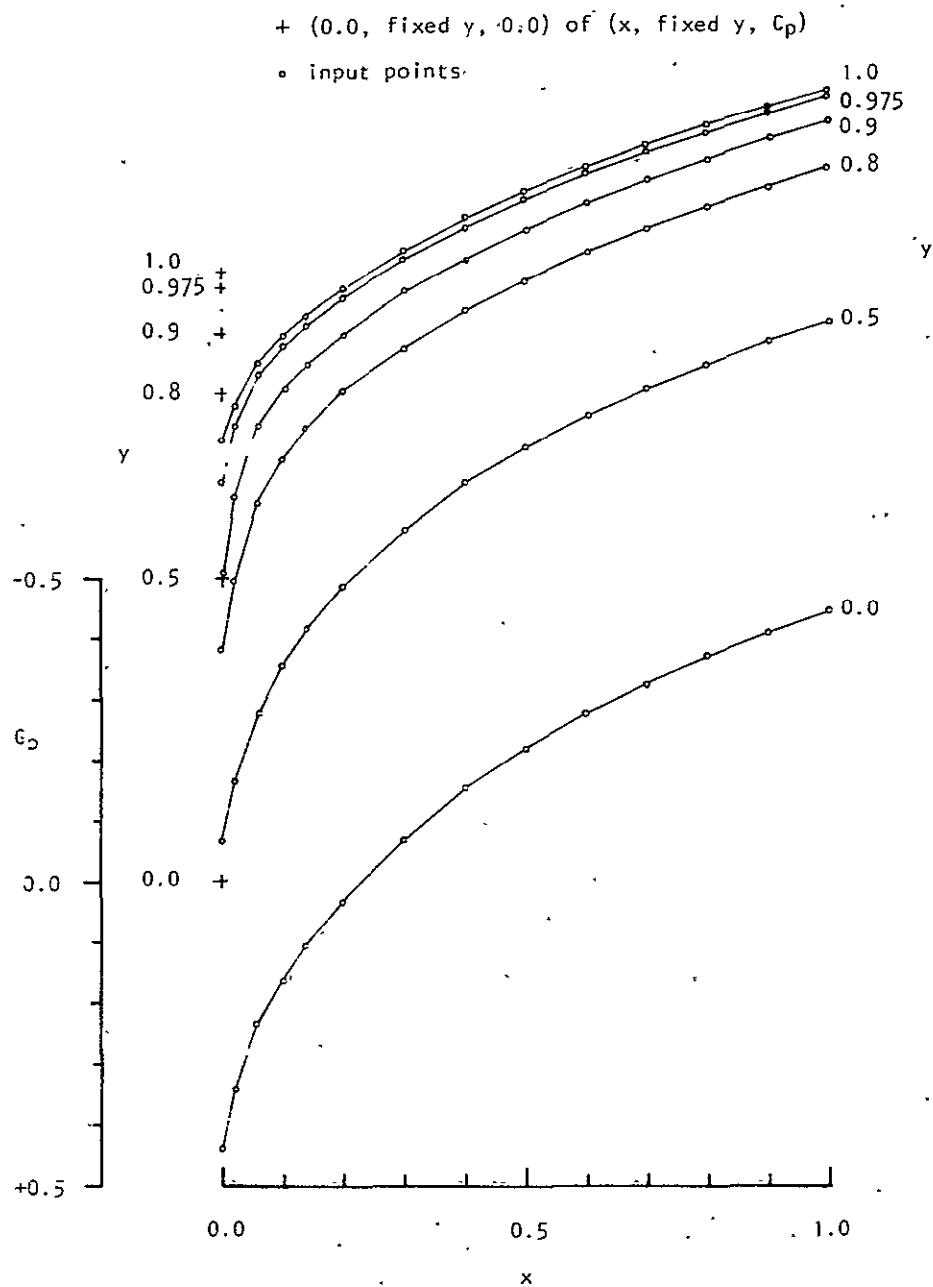


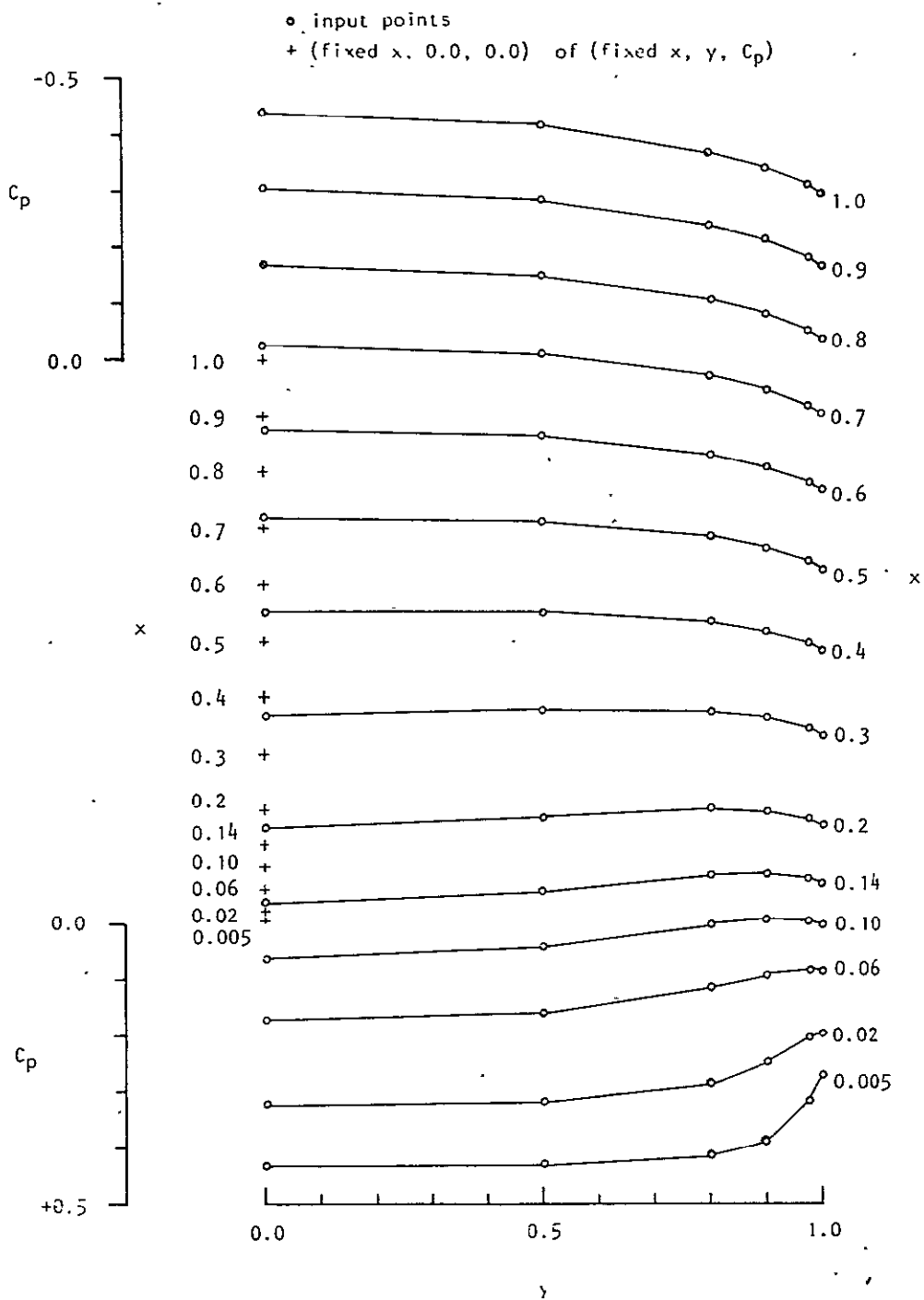
Figure 4. - Concluded.



(a) Chordwise distribution.

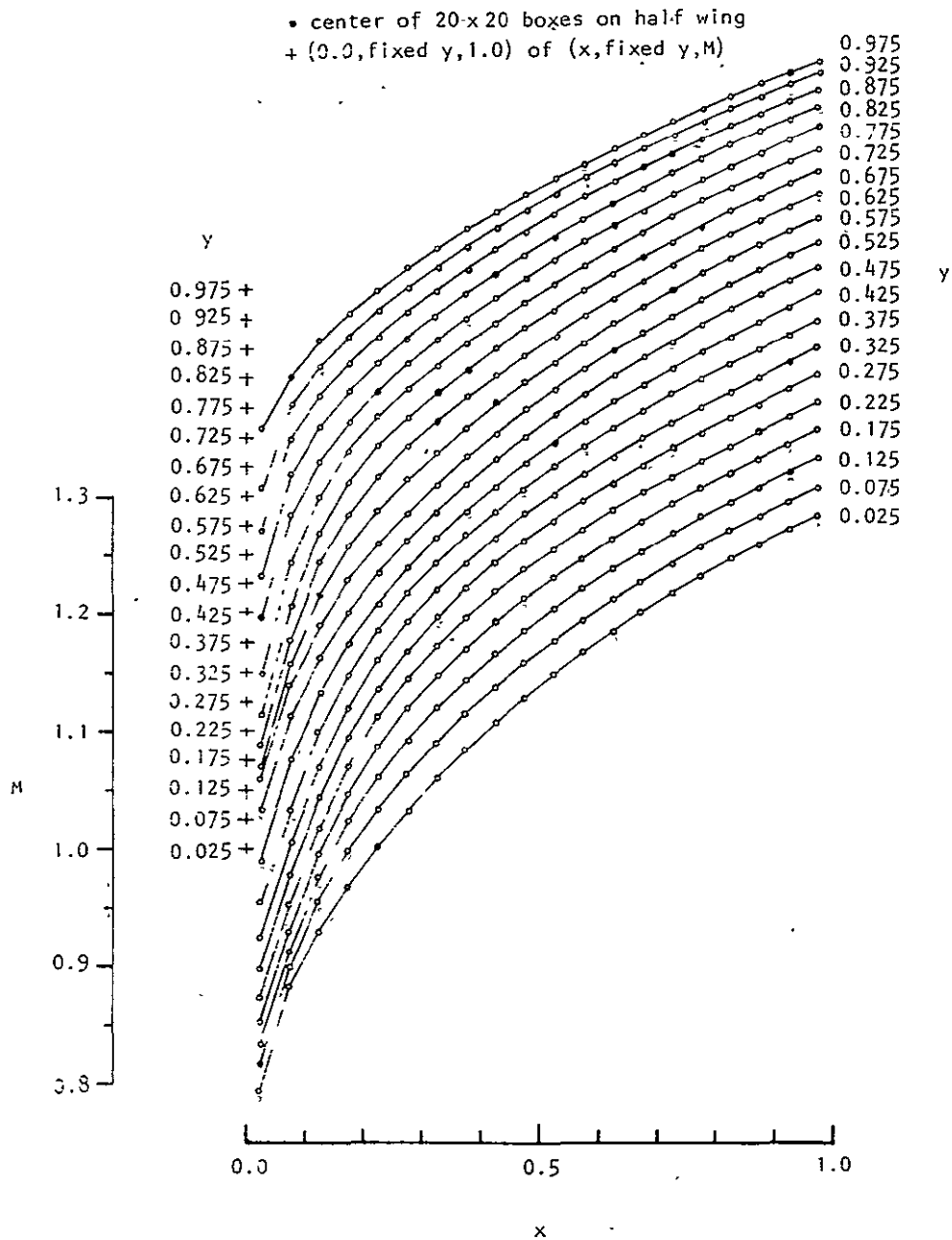
Figure 5. - Input steady state (mean) pressure coefficient on rectangular wing of aspect ratio 2.0 and thickness ratio 0.0521 (reference 8).

ORIGINAL PAGE IS  
OF POOR QUALITY



(b) Spanwise direction.

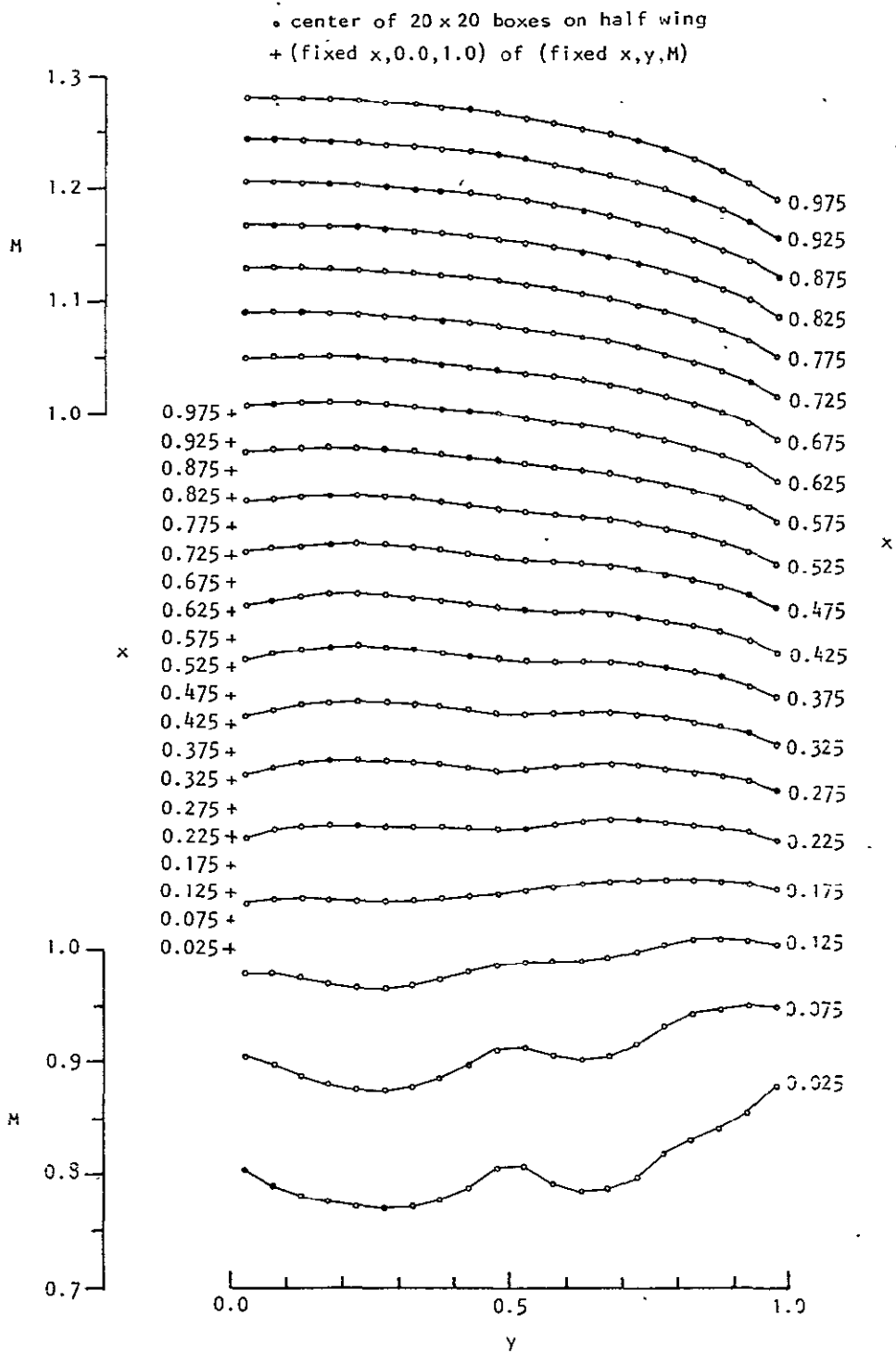
Figure 5. - Concluded.



(a) Chordwise distribution.

Figure 6. - Computed steady state (mean) Mach number on rectangular wing of aspect ratio 2.0 and the thickness ratio 0.0521.





(b) Spanwise distribution.

Figure 6. - Concluded.

## APPENDIX

### COMPUTER PROGRAM

The computer program listed in this appendix is dimensioned to handle a maximum of 30 boxes, either in chordwise or spanwise directions, in approximating the wing planform. The maximum numbers of leading and trailing edge segments are, respectively, 7 and 2 per semispan. The program can handle up to 3 wing deflection mode shapes. The maximum number of points used in spline-surface fitting is 100, so the maximum number of input points to describe the wing deflections and the Mach number or pressure coefficient distribution is also 100. These limitations can easily be increased by changing the dimensions of the corresponding variables in the computer program.

In order to activate the smoothing option in two-dimensional cubic-spline data fitting, it is required that a two-digit fixed point number be assigned to the last argument, NSMOS, of subroutine SPLN1. The right digit is for the control of the number of smoothings desired; and the left-digit is for pre-interpolation, zero for omitting pre-interpolation and non-zero for including pre-interpolation. The pre-interpolation is a process to increase the number of known points to be used in the interpolation by inserting an additional point between every successive pair of input points in the original set.

## Input Guide

Data are input through the subroutine DATRD using the one dimensional array DA with a size of 1005. The allowable maximum number for some of the input data as indicated below may be changed if the dimensions of the corresponding storage array and computational operations are also changed accordingly. Subroutine DATRD initializes DA(1) through DA(22) to blank, the weighting factors in DA(104), DA(108), ---, DA(500) to 1.0, and the remaining portion of the DA array to 0.0.. Consequently, these are the default values. The layout of the array DA(k) as it is presently used is similar to that in reference 4 and is as follows:

- 1-7: Title
- 8-12: Not used
- 13-19: Mode title
- 20-22: Not used
- 23: Frequency, (cycle/sec)
- 24: Overall length of wing in streamwise direction, (ft or meter)
- 25: Speed of sound of the freestream, (ft/sec or meter/sec)
- 26: (0) - indicates the frequency is the first one for a new wing  
(1) - indicates the frequency is the additional one for the same wing
- 27: Number of boxes in streamwise direction (maximum 30)
- 28: Number of deflection modes (maximum 3)
- 29: Number (m) of segments of leading edge per semispan to be given, excluding segment from origin to  $y_0$  ( $m_{\max} = 7$ )
- 30-44: Coordinates of points on the leading edge, (ft or meter)  
(in sequence of  $y_0, x_1, y_1, x_2, y_2, \dots, x_m, y_m$ ),  $m_{\max} = 7$
- 45: Number (n) of segments of trailing edge per semispan to be given, (default: unswept trailing edge),  $n_{\max} = 2$
- 46-48: Coordinates of points on the trailing edge, (ft or meter)  
(in sequence of  $x_0, y_1, x_1$  for  $n = 2$ ,  
or  $x_0$  (only) for  $n = 1$ ,  
no input for  $n = 0$ ;  
last trailing edge point coincides with the last leading edge point and is set internally)

- 49: Number of boxes allowed for upstream influence (if this location is left blank or assigned a zero, it will assume  $DA(49)=DA(27)$  and in no case  $DA(49)>DA(27)$  is allowed).
- 50: (0) - indicates to calculate cases with and without thickness effect
  - (1) - indicates to calculate case without thickness effect only
  - (2) - indicates to calculate case with thickness effect only
- 51: Indicator to suppress calculation of potential for a mode
  - (0) - no suppression
  - (1) - suppression
- 52-53: Coefficients of the deflection polynomial (in the sequence of  $a_0$  and  $a_1$ )
- 54-70: Not used\*
- 71-72: Coefficients of the Mach number distribution polynomial (in the sequence of  $a_0$  and  $a_1$ )
- 73-95: Not used\*
- 96: Indicator of the type of wing thickness effect input
  - (1) - pressure coefficient
  - (2) - Mach number
- 97: Number of points at which pressure coefficient or Mach number to be given
- 98: Number of points on which deflections to be given
- 99-100: Not used\*
- 101-500: Deflection data for a maximum of 100 points (in the sequence of  $x$ ,  $y$ , deflection and weighting factor)
- 501-700: Not used\*\*
- 701-1000: Pressure coefficient or Mach number data for a maximum of 100 points (in the sequence of  $x$ ,  $y$  and pressure coefficient or Mach number)

The remaining part of DA array is used for the control of intermediate results print out. When the latter is desired, a non-zero positive integer number should be entered at locations in the DA array corresponding to the information from the one particular subroutine that is needed.

- 1001: CBA: for wing deflection (DRED)  
       A=1, for NEW=1 } spline-surface fitted results  
       B=1, for NEW=2 }  
       C - not applicable
- 1002: BA: for wing upwash (WVAL)  
       A=1, - upwash  
       B - not applicable
- 1003: FEDCBA for velocity potential (BOXP and BOXPO)  
       A=1, for NEW=1 } velocity potential  
       B=1, for NEW=2 }  
       C=1, for NEW=1 } influence coefficient and solution matrices  
       D=1, for NEW=2 }  
       E=1, for NEW=1 } pressure coefficient  
       F=1, for NEW=2 }
- 1004: A: for Mach number (MRED)  
       A=1 - spline-surface fitted results
- 1005: DCBA: for wing shape (SHAPE and PLNFM)  
       A=1, for NEW=1 } distributions of box, box area, leading and  
       B=1, for NEW=2 } trailing edges  
       C=1 - Mach number at box centers  
       D - not applicable

The format of the input data card is (A1, A5, I6, 6A10, A8). The first field is for the control of clearing the data array, DA, for a new wing (+) and the control to indicate the end of the set of data (-). The second field is the indicator for the type of data, either numeric (blank) or alphameric (ALPHA). The third field is the designator for the relative location in the data array of the first number to follow in the fourth field. If this field is left blank, or a zero is entered, the execution will be terminated. The fourth and fifth fields are for five consecutive input data each occupying 12 columns plus 8 blank columns at the end. All the fixed point numbers are right-adjusted and the decimal point for the floating point number must be included. If an input datum is left blank, no change at the storage location for that particular datum in the data array will occur unless the set of the input data is for a new wing.

Those storages currently not used in array DA marked with \* are reserved for future improvements in the method used for the functional form of data input. Those marked with \*\* are reserved for the case where large numbers of data input points for deflections or Mach numbers are required.

## Sample Input

A typical input data deck set-up for an aspect ratio 1.5 delta wing having an elliptic lateral cross-section with 10% thickness ratio performing plunge and pitch about its apex are given below.

The input format is (A1, A5, I6, 6A10, A8).

Card 1: title of the case under consideration.

Card 2: title of the first mode of deflection.

Card 3: first frequency (cycle/sec), centerline chord length (ft), reference velocity (ft/sec).

Card 4: number of boxes along the centerline chord, number of deflection modes, number of total leading edge segments of the wing.

Card 5: spanwise coordinate (ft) of the first section of the leading edge, chordwise and spanwise coordinates (ft) of the next section (the sequence is  $y_0, x_1, y_1$  -- e.g., see figure 1).

Card 6: first mode of deflection  $f = 1.0$   
the " - " sign indicates the end of the group of data cards to be read at this stage.

Card 7: title of the second mode.

Card 8: second mode of deflection  $f = 0.1x$ .

Card 9: identification of the type of input regarding the wing thickness effect (Mach number for this case), number of points on the wing this information to be given.

Cards 10 to 69:

chordwise and spanwise coordinates (ft) of a point on the wing, and the Mach number at this point.

the " - " sign on the last card indicates the end of the group of data cards to be read at this stage.

Cards 70 and 71:

additional frequencies for the same wing, one card is read in at one time.

Card 72: blank card to make an exit from the computer.

The card images are as follows:

1	2	3	4	5	6	7	8
1234567890123456789012345678901234567890123456789012345678901234567890							
ALPHA	ASPECT RATIO 1.5 DELTA WING (TAU=0.10)						1
ALPHA	13PLUNGE						2
23	0.159154941	10.0	1000.0				3
27	30	2	1				4
30	0.0	10.0	3.75				5
52	1.0						6
ALPHA	13PITCH ABOUT ROOT LEADING EDGE X=0.0						7
52	0.0	0.1					8
96	1	60					9
701	0.1		0.140232				10
704	0.3		0.135125				11
707	0.5		0.132175				12
710	0.7		0.129943				13
713	0.9		0.128084				14
716	1.3		0.124980				15
719	1.7		0.122340				16
722	2.1		0.119963				17
725	2.5		0.117746				18
728	2.9		0.115627				19
731	3.3		0.113564				20
734	3.9		0.110510				21
737	4.5		0.107428				22
740	5.1		0.104243				23
743	5.7		0.100872				24
746	6.3		0.097215				25
749	6.9		0.093129				26
752	7.5		0.088395				27
755	7.9		0.084697				28
758	8.3		0.080342				29
761	8.7		0.074970				30
764	9.1		0.067825				31
767	9.3		0.063062				32
770	9.5		0.056823				33
773	9.7		0.047661				34
776	9.9		0.029651				35
779	1.7	0.5	0.122340				36
782	3.3	1.0	0.113564				37
785	4.5	1.0	0.107428				38
788	5.7	1.0	0.100872				39
791	6.9	1.0	0.093129				40
794	8.3	1.0	0.080342				41
797	9.5	1.0	0.056823				42
800	4.5	1.5	0.107428				43
803	6.3	2.0	0.097215				44
806	7.5	2.0	0.088395				45
809	8.7	2.0	0.074970				46
812	9.7	2.0	0.047661				47
815	8.3	2.5	0.080342				48
818	9.7	3.0	0.047661				49
821	2.5	0.5	0.117746				50
824	3.9	0.5	0.110510				51
827	5.1	0.5	0.104243				52
830	6.3	0.5	0.097215				53
833	7.5	0.5	0.088395				54
836	9.1	0.5	0.067825				55
839	9.9	0.5	0.029651				56
842	5.1	1.5	0.104243				57
845	6.3	1.5	0.097215				58
848	7.9	1.5	0.084697				59
851	9.1	1.5	0.067825				60
854	9.9	1.5	0.029651				61
857	5.7	2.0	0.100872				62
860	6.9	2.5	0.093129				63
863	9.5	2.5	0.056823				64
866	9.9	2.5	0.029651				65
869	8.3	3.0	0.080342				66
872	9.1	3.0	0.067825				67
875	9.5	3.5	0.056823				68
878	9.9	3.5	0.029651				69
23	1.591549407						70
23	6.366197628						71
							72

ORIGINAL PAGE IS  
OF POOR QUALITY



ORIGINAL PAGE IS  
OF POOR QUALITY

# Sample Output

ASPECT RATIO 1.5 DELTA WING (TAU=0.10)

30 BOXES ALONG ROOT CHORD

ROOT CHORD LENGTH = 10.00 FT

REDUCED FREQUENCY = .010

FREE STREAM VELOCITY = 1000.00 FT/SEC

FREQUENCY = 1.592E-01 CYCLE/SEC

MODE NO. 1 PLUNGE

MODE NO. 2 PITCH ABOUT ROOT LEADING EDGE X=0.0

## GENERALIZED FORCES (NO THICKNESS EFFECT)

Modes	PPES.	DEFL.	REAL PART	IMAG PART	ABS. VALUE	PHASE ANGLE
1	1		2.15396E-06	-2.42783E-02	2.42783E-02	-89.9949
1	2		-1.69639E-07	-1.61159E-02	1.61159E-02	-90.0006

## GENERALIZED FORCES (NO THICKNESS EFFECT)

Modes	PPES.	DEFL.	REAL PART	IMAG PART	ABS. VALUE	PHASE ANGLE
2	1		-2.42788E+00	-2.44689E-02	2.42800E+00	-179.4226
2	2		-1.61162E+00	-1.81349E-02	1.61173E+00	-179.3553

MODE NO. 1

MODE NO. 2

## GENERALIZED FORCES (WITH THICKNESS EFFECT)

Modes	PPES.	DEFL.	REAL PART	IMAG PART	ABS. VALUE	PHASE ANGLE
1	1		-1.00531E-05	-2.43535E-02	2.43535E-02	-90.0237
1	2		-1.29338E-05	-1.61534E-02	1.61534E-02	-90.0459

## GENERALIZED FORCES (WITH THICKNESS EFFECT)

Modes	PPES.	DEFL.	REAL PART	IMAG PART	ABS. VALUE	PHASE ANGLE
2	1		-2.43541E+00	-2.33020E-02	2.43552E+00	-179.4518
2	2		-1.61539E+00	-1.68848E-02	1.61548E+00	-179.4011

ASPECT RATIO 1.5 DELTA WING (TAU=0.10)

30 BOXES ALONG ROOT CHORD

ROOT CHORD LENGTH = 10.00 FT

REDUCED FREQUENCY = .100

FREE STREAM VELOCITY = 1000.00 FT/SEC

FREQUENCY = 1.592E+00 CYCLE/SEC

MODE NO. 1

MODE NO. 2

#### GENERALIZED FORCES (NO THICKNESS EFFECT)

MODES		REAL PART	IMAG PART	ABS. VALUE	PHASE ANGLE
PRES.	DEFL.				
1	1	3.64268E-04	-2.42141E-01	2.42141E-01	-89.9138
1	2	1.20006E-04	-1.60604E-01	1.60604E-01	-89.9572

#### GENERALIZED FORCES (NO THICKNESS EFFECT)

MODES		REAL PART	IMAG PART	ABS. VALUE	PHASE ANGLE
PRES.	DEFL.				
2	1	-2.42556E+00	-2.45153E-01	2.43792E+00	-174.2287
2	2	-1.60948E+00	-1.81827E-01	1.61972E+00	-173.5545

MODE NO. 1

MODE NO. 2

#### GENERALIZED FORCES (WITH THICKNESS EFFECT)

MODES		REAL PART	IMAG PART	ABS. VALUE	PHASE ANGLE
PRES.	DEFL.				
1	1	-7.33272E-04	-2.42506E-01	2.42507E-01	-90.1732
1	2	-1.03563E-03	-1.60598E-01	1.60601E-01	-90.3695

#### GENERALIZED FORCES (WITH THICKNESS EFFECT)

MODES		REAL PART	IMAG PART	ABS. VALUE	PHASE ANGLE
PRES.	DEFL.				
2	1	-2.43011E+00	-2.34356E-01	2.44138E+00	-174.4915
2	2	-1.61039E+00	-1.70175E-01	1.61936E+00	-173.9678

ORIGINAL PAGE IS  
OF POOR QUALITY

ASPECT RATIO 1.5 DELTA WING (TAU=0.10)

30 BOXES ALONG ROOT CHORD

ROOT CHORD LENGTH = 10.00 FT

REDUCED FREQUENCY = .400

FREE STREAM VELOCITY = 1000.00 FT/SEC

FREQUENCY = 6.366E+00 CYCLE/SEC

MODE NO. 1

MODE NO. 2

GENERALIZED FORCES (NO THICKNESS EFFECT)

MODES			REAL PART	IMAG PART	ABS. VALUE	PHASE ANGLE
PRES.	DEFL.					
1	1		2.60265E-02	-9.51379E-01	9.51735E-01	-89.4330
1	2		1.93704E-02	-6.28441E-01	6.28740E-01	-88.2345

GENERALIZED FORCES (NO THICKNESS EFFECT)

MODES			REAL PART	IMAG PART	ABS. VALUE	PHASE ANGLE
PRES.	DEFL.					
2	1		-2.41529E+00	-9.98983E-01	2.61373E+00	-157.5296
2	2		-1.60030E+00	-7.44074E-01	1.76483E+00	-155.0635

MODE NO. 1

MODE NO. 2

GENERALIZED FORCES (WITH THICKNESS EFFECT)

MODES			REAL PART	IMAG PART	ABS. VALUE	PHASE ANGLE
PRES.	DEFL.					
1	1		1.95356E-02	-9.43362E-01	9.43564E-01	-88.8137
1	2		1.17730E-02	-6.19079E-01	6.19191E-01	-88.9105

GENERALIZED FORCES (WITH THICKNESS EFFECT)

MODES			REAL PART	IMAG PART	ABS. VALUE	PHASE ANGLE
PRES.	DEFL.					
2	1		-2.39926E+00	-9.75339E-01	2.58993E+00	-157.8775
2	2		-1.59205E+00	-7.16462E-01	1.73672E+00	-155.6357

# Program Listing

	1	TAPE5=INPUT,TAPE6=OUTPUT)	SBOXR2	2
		MAI=...	SBOXR2	3
		MAXIMUM NUMBER OF BOXES ALONG THE CENTERLINE CHORD ALLOWED	SBOXR2	4
		EXTENSION STATEMENT	SBOXR2	5
		MAXIMUM NUMBER OF MODES ALLOWED IN DIMENSION STATEMENT	SBOXR2	6
		MAXIMUM NUMBER OF BOXES ALLOWED FOR SPLINE FITTING	SBOXR2	7
		NUMBER OF BOXES ALLOWED IN DIMENSION STATEMENT	SBOXR2	8
		DATA 2/10/7	A	2
		ML(30),MLH(30),MLT(2,30),MLC(2,30),MB,AMA(30,30)	A	3
		XEDG(8),YEDG(8),NEDGI,NS,XTDG(3),YTDG(3),NTDGI,NST	A	4
		SFMY(100),SFMY(100),SFMH(103),KSFM,T(103,104),NB,NM	A	5
		DA(100),AREA,CK,D,DH,UI,YMAX,JMAX,JHAK,IEDG,IW,L,NEW	A	6
		DIMENSION A(2,60,30),S(2,30,30)	SBOXR2	10
		SFDA(100,3),SFDY(100,3),SFDH(103,3),KSFD(3),G(3)	SBOXR2	11
		DATA 2/10/7	SBOXR2	12
		IW=0	SBOXR2	13
		IW=0	SBOXR2	14
		NEDGI=0	SBOXR2	15
		NTDGI=3	SBOXR2	16
		MB=30	SBOXR2	17
		MD=3	SBOXR2	18
		NB=100	SBOXR2	19
		MATCH ABOVE NUMBERS TO THE DEFINED DIMENSION IN STORAGE ARRAYS	SBOXR2	20
		ME=MB+1	SBOXR2	21
		MC=2*MB	SBOXR2	22
		NM=NB+3	SBOXR2	23
		WRITE(IW,55)	SBOXR2	24
		READ DATA FOR THE ACTUAL WING	SBOXR2	25
	100	CALL DATRD(DA)	SBOXR2	26
		NEW=1	SBOXR2	27
		TEST1=DA(1001)+DA(1002)+DA(1003)+DA(1004)+DA(1005)	SBOXR2	28
		CK=DA(23)*DA(24)/DA(25)+6.28318531	SBOXR2	29
		DI=DA(27)	SBOXR2	30
		L=DI	SBOXR2	31
		D=1.0/DI	SBOXR2	32
		DH=0.5*D	SBOXR2	33
		WRITE(IW,60) (DA(I),I=1,7)	SBOXR2	34
	110	IF (L) 600,600,120	SBOXR2	35
	120	IF (MB-L) 600,130,130	SBOXR2	36
	130	WRITE(IW,65) L,DA(24),CK,DA(25),DA(23)	SBOXR2	37
		IF (DA(26)) 160,150,160	SBOXR2	38
	150	CALL SHAPE	SBOXR2	39
		IF(NEW.EQ.2) GO TO 180	SBOXR2	40
		AC=8.0/AREA	SBOXR2	41
	160	LIM=ML(L)	SBOXR2	42
		IF (LIM-MB) 170,170,650	SBOXR2	43
	170	LIM2=2*MB	SBOXR2	44
		NFLNS = DA(49)	SBOXR2	45
		IF(NFLNS.EQ.0) NFLNS = L	SBOXR2	46
		LPOT = MINO(L,NFLNS)	SBOXR2	47
		LIM1=2*MB	SBOXR2	48
		CALL POT2(LIM1,LIM2,LPOT,CK,D,A)	SBOXR2	49
	180	CONTINUE	SBOXR2	50
		M=0	SBOXR2	51
		K=DA(28)	SBOXR2	52
		GO TO 230	SBOXR2	53
			SBOXR2	54
		PRELIMINARY CALCULATIONS ARE FINISHED.	SBOXR2	55
		THE NEXT SECTION IS GONE THROUGH FOR EACH MODE.	SBOXR2	56
			SBOXR2	57
	200	IF (DA(26)) 230,210,230	SBOXR2	58
	210	IF(NEW.EQ.2) GO TO 230	SBOXR2	59
		CALL DATRD(DA)	SBOXR2	60

230 K=K-1	SBOXR2	61
M=M+1	SBOXR2	62
IF (TEST1.LT.1.0) GO TO 250	SBOXR2	63
WRITE (1W,55)	SBOXR2	64
250 WRITE (1W,15) M	SBOXR2	65
WRITE (1W,16) (DA(1),1=13,19)	SBOXR2	66
15 FORMAT (1H0,15X,8HMODE NO.,13)	SBOXR2	67
16 FORMAT (1H+,30X,7A10)	SBOXR2	68
IF (DA(26)) 250,280,290	SBOXR2	69
280 IF (NEW.EQ.2) GO TO 290	SBOXR2	70
IPRINT=DA(1001)	SBOXR2	71
CALL DRED(SFDX,SFDY,SFDH,KSF0,SFMX,SFMY,SFMH,KSFH,DA,T	SBOXR2	72
,XX,YY,IW,L,HLC,H,MD,NB,NEW,NH,KSFS,IPRINT)	SBOXR2	73
G(M)=DA(51)	SBOXR2	74
290 IF (K) 310,310,300	SBOXR2	75
300 IF (M=MD) 200,310,310	SBOXR2	76
310 CONTINUE	SBOXR2	77
IF (I+1X(DA(50)).E2.2.AND.NEW.EQ.1) GO TO 490	SBOXR2	78
DO 470 M1=1,M	SBOXR2	79
IF (J(M1)) 480,320,480	SBOXR2	80
320 CONTINUE	SBOXR2	81
IPRINT=DA(1002)	SBOXR2	82
CALL XVAL(XX,YY,ATDG,YTDG,XTE,SFDX,SFDY,SFDH,KSF0,S,D,CK	SBOXR2	83
,I,L,ML,M1,NEW,NST,MB,MD,NB,NH,KSFS,IPRINT)	SBOXR2	84
IF (YDGG(1)) 590,300,330	SBOXR2	85
330 YDGG(1)=1	SBOXR2	86
CCF=NEW PCGE CORRECTION	SBOXR2	87
CC=1.0	SBOXR2	88
350 C(J,1)=C(J,1)+C(J,2,1)*.973.14159265	SBOXR2	89
360 CONTINUE	SBOXR2	90
YD=YD(1)	SBOXR2	91
YD=DA(4)	SBOXR2	92
IF (I+1X(DA(1003))	SBOXR2	93
CALL XVAL(XX,YY,ATDG,YTDG,XTE,XLE,A,AR,DAN,T,EDG,S,CK,D,YE	SBOXR2	94
,IPRINT,MLT,MLC,AMA	SBOXR2	95
,JMAK,C,ML,MLC,M1,NEW,NST,IH,MB,MC,ME,MD)	SBOXR2	96
370 CONTINUE	SBOXR2	97
380 CONTINUE	SBOXR2	98
390 CONTINUE	SBOXR2	99
400 CONTINUE	SBOXR2	100
410 CONTINUE	SBOXR2	101
420 CONTINUE	SBOXR2	102
430 CONTINUE	SBOXR2	103
440 CONTINUE	SBOXR2	104
450 CONTINUE	SBOXR2	105
460 CONTINUE	SBOXR2	106
470 CONTINUE	SBOXR2	107
480 CONTINUE	SBOXR2	108
490 CONTINUE	SBOXR2	109
500 CONTINUE	SBOXR2	110
510 CONTINUE	SBOXR2	111
520 CONTINUE	SBOXR2	112
530 CONTINUE	SBOXR2	113
540 CONTINUE	SBOXR2	114
550 CONTINUE	SBOXR2	115
560 CONTINUE	SBOXR2	116
570 CONTINUE	SBOXR2	117
580 CONTINUE	SBOXR2	118
590 CONTINUE	SBOXR2	119
600 CONTINUE	SBOXR2	120
610 CONTINUE	SBOXR2	121
620 CONTINUE	SBOXR2	122
630 CONTINUE	SBOXR2	123
640 CONTINUE	SBOXR2	124
650 CONTINUE	SBOXR2	125
660 CONTINUE	SBOXR2	126
670 CONTINUE	SBOXR2	127
680 CONTINUE	SBOXR2	128
690 CONTINUE	SBOXR2	129
700 CONTINUE	SBOXR2	130

450	WRITE(IW,30) M1,M2,S1,S2,S3,S4	SBOXR2	131
30	FORMAT(1H02I6,1P3E19.5,0P1F16.4)	SBOXR2	132
470	CONTINUE	SBOXR2	133
480	WRITE(IW,55)	SBOXR2	134
490	CONTINUE	SBOXR2	135
	IF(DA(26).GT.C.) GO TO 510	SBOXR2	136
	DO 500 I=13,22	SBOXR2	137
500	DA(I)=Z	SBOXR2	138
510	IF((FIX(DA(50))-1.EQ.0) GO TO 100	SBOXR2	139
	IF(NEW.EQ.2) GO TO 100	SBOXR2	140
	NEW=C	SBOXR2	141
	IF(DA(26)) 18C,580,180	SBOXR2	142
580	IPRINT=DA(1G04)	SBOXR2	143
	CALL MRED(DA,T,NH,NB,KSFH,SFMX,SFMY,SFMH,IW,IPRINT)	SBOXR2	144
	GO TO 150	SBOXR2	145
C		SBOXR2	146
C	ERROR EXISTS	SBOXR2	147
C		SBOXR2	148
590	IPR=30	SBOXR2	149
	GO TO 610	SBOXR2	150
600	IPR=27	SBOXR2	151
610	WRITE(IW,35) IPR	SBOXR2	152
35	FORMAT(1H010X,8HBAD DATA14)	SBOXR2	153
	GO TO 700	SBOXR2	154
650	WRITE(IW,4C)	SBOXR2	155
40	FORMAT(1H010X,42HLATERAL LIMIT ON NUMBER OF BOXES EXCEEDED.)	SBOXR2	156
700	STOP	SBOXR2	157
55	FORMAT(1H1)	SBOXR2	158
60	FORMAT(1H0,10X,7A10)	SBOXR2	159
65	FORMAT(1H0,10X,12,23H BOXES ALONG ROOT CHORD,	SBOXR2	160
1	15X,22HROOT CHORD LENGTH =,F8.2,3H FT/	SBOXR2	161
2	1H0,10X,19HREDUCED FREQUENCY =,F6.3,	SBOXR2	162
3	15X,22HFREE STREAM VELOCITY =,F8.2,7H FT/SEC/	SBOXR2	163
4	1H0,10X,11HFREQUENCY =,1PE11.3,10H CYCLE/SEC)	SBOXR2	164
	END	SBOXR2	165

LINE	STATEMENT	DATE	TIME
1	SUBROUTINE DATRO(DATA)	2	
2	CARD-READ SUBROUTINE *DATRO(DATA(I))*	3	
3	DIMENSION DRBU(12),DATA(1),DDRBU(10)	4	
4	DATA ATEST,HALPHA/,DTEST/1H /,ETEST/1H-/ ,STEST/1H-/	5	
5	DATA 2/1H /	6	
6	IR=5	7	
7	IN=6	8	
8	1 READ (IR,2) EMIN,ALP,IND,(DRBU(I),I=1,7)	9	
9	2 FORMAT(1A5,I6,6A10,A8)	10	
10	IF(IND.EQ.0) GO TO 20	11	
11	IF (EMIN.NE.STEST) GO TO 105	12	
12	NEW MING IF COLUMN 1 CONTAINS A PLUS SIGN	13	
13	INITIALIZATION OF DATA ARRAY	14	
14	DO 101 I=23,1005	15	
15	101 DATA(I)=0.0	16	
16	DO 102 I=1,22	17	
17	102 DATA(I)=Z	18	
18	DO 103 I=104,700,4	19	
19	103 DATA(I)=1.0	20	
20	105 CONTINUE	21	
21	IF (ALP.EQ.ATEST) GO TO 9	22	
22	IF (ALP.NE.DTEST) GO TO 8	23	
23	NUMERIC CARD	24	
24	DO 3 I=1,6	25	
25	DDRBU(I)=DATA(I)	26	
26	CONTINUE	27	
27	DECOO=(10,190,DDRBU(I),I=1,5)	28	
28	DO 5 I=1,5	29	
29	IF(DRBU(I))4,2,4	30	
30	TEST FOR BLANK FIELD	31	
31	IF(DRBU(I))4,2,4	32	
32	DATA(IND)=DATA(I)	33	
33	IN=IN+1	34	
34	GO TO 11	35	
35	20-2 244	36	
36	DO 17 I=1,7	37	
37	DATA(IND)=DATA(I)	38	
38	IND=IND+1	39	
39	IF (EMIN.NE.DTEST) GO TO 1	40	
40	RETURN IF COLUMN 1 CONTAINS A MINUS SIGN	41	
41	20-2 244	42	
42	20-2 244	43	
43	20-2 244	44	
44	20-2 244	45	
45	20-2 244	46	
46	20-2 244	47	
47	20-2 244	48	
48	20-2 244	49	
49	20-2 244	50	
50	20-2 244	51	
51	20-2 244	52	
52	20-2 244	53	
53	20-2 244	54	
54	20-2 244	55	
55	20-2 244	56	
56	20-2 244	57	

		SHAPE	2
		A	2
		A	3
		A	4
		A	5
		A	6
		SHAPE	4
		SHAPE	5
		SHAPE	6
		SHAPE	7
		SHAPE	8
		SHAPE	9
		SHAPE	10
		SHAPE	11
		SHAPE	12
		SHAPE	13
		SHAPE	14
		SHAPE	15
		SHAPE	16
100	IF (NS-NEUGI+1) 105,105,850	SHAPE	17
105	NSP=NS+1	SHAPE	18
	NS=NSP	SHAPE	19
	IF (DA(24)) 855,855,110	SHAPE	20
110	DO 115 I=1,NSP	SHAPE	21
	XEDG(I)=DA(2*I+27)/DA(24)	SHAPE	22
115	YEDG(I)=DA(2*I+28)/DA(24)	SHAPE	23
	XEDG(I)=0.0	SHAPE	24
	NST=DA(45)	SHAPE	25
	IF (NST) 871,125,130	SHAPE	26
125	XTDG(I)=1.0	SHAPE	27
	NST=1	SHAPE	28
	GO TO 141	SHAPE	29
130	IF (NST-NTDGI+1) 135,135,871	SHAPE	30
135	NSTP=NST	SHAPE	31
	DO 140 I=1,NSTP	SHAPE	32
	YTDG(I)=DA(2*I+43)/DA(24)	SHAPE	33
140	XTDG(I)=DA(2*I+44)/DA(24)	SHAPE	34
141	NSTP=NST+1	SHAPE	35
	NST=NSTP	SHAPE	36
	XTDG(NSTP)=XEDG(NSP)	SHAPE	37
	YTDG(NSTP)=YEDG(NSP)	SHAPE	38
	YTDG(I)=0.0	SHAPE	39
	IF (NST.EQ.NTDGI) GO TO 150	SHAPE	40
	JL=NST+1	SHAPE	41
	DO 142 I=JL,NTDGI	SHAPE	42
	XTDG(I)=XTDG(I-1)	SHAPE	43
142	YTDG(I)=YTDG(I-1)	SHAPE	44
150	IF (ABS(XTDG(NSTP)-XEDG(NSP)+YTDG(NSTP)-YEDG(NSP)).GT.ERRR)	SHAPE	45
	S GO TO 871	SHAPE	46
	GO TO 550	SHAPE	47
C		SHAPE	48
C	FOR WING WITH THICKNESS	SHAPE	49
C	COMPUTE MACH NUMBER DISTRIBUTION ON PHYSICAL WING	SHAPE	50
200	DM=0.	SHAPE	51
	DO 210 I=1,NB	SHAPE	52
	DO 210 J=1,NB	SHAPE	53
210	AMA(J,I)=1.0	SHAPE	54
	DO 215 I=1,L	SHAPE	55
	IF (MLC(1,I).EQ.0) GO TO 215	SHAPE	56
	J=MLC(1,I)	SHAPE	57
	K=MLC(2,I)	SHAPE	58
	CALL SURF2(XX(I),YY,J,K,1,AMA(1,I),DM,DM,SFMX,SFMY,SFMH,KSFH,1)	SHAPE	59
215	CONTINUE	SHAPE	60
	IF (J.KITE.EQ.0) GO TO 235	SHAPE	61
C	PRINT OUT MACH DISTRIBUTION	SHAPE	62
	WRITE(IH,60)	SHAPE	63
	DO 230 I=1,L	SHAPE	64
	IF (MLC(1,I).EQ.0) GO TO 230	SHAPE	65



		SHAPE	
	JL=MLC(2,I)-MLC(1,I)+1	SHAPE	66
	IF(JL.EQ.0) GO TO 230	SHAPE	67
	WRITE(IH,70) I	SHAPE	68
	JLP=JL/6	SHAPE	69
	IF(JL-6*JLP.NE.0) JLP=JLP+1	SHAPE	70
	K1=MLC(1,I)-1	SHAPE	71
	JL=MLC(2,I)	SHAPE	72
	DO 225 J=1,JLP	SHAPE	73
	K=K1+J	SHAPE	74
	225 WRITE(IH,80) ((J1,AMA(J1,I)),J1=K,JL,JLP)	SHAPE	75
	230 CONTINUE	SHAPE	76
C		SHAPE	77
	235 CONTINUE	SHAPE	78
	GO TO 750	SHAPE	79
	550 Y1=YEDG(I)	SHAPE	80
	YMAX=YEDG(NSP)	SHAPE	81
	IF(Y1.LT.0.0) GO TO 860	SHAPE	82
	AREA=0.0	SHAPE	83
	XX(I)=DH	SHAPE	84
	YY(I)=DH	SHAPE	85
	DO 560 I=2,MB	SHAPE	86
	YY(I)=YY(I-1)+D	SHAPE	87
	560 XX(I)=XX(I-1)+C	SHAPE	88
C		SHAPE	89
C	CALCULATE PHYSICAL FULL WING AREA AND ML(NEW,I)	SHAPE	90
C	COMPUTE AND IDENTIFY BOX DISTRIBUTION ON THE WING PLANFORM	SHAPE	91
	565 CALL PLNFH(XEDG,YEDG,XTOG,YTOG,XX,YY,XLE,XTE,AR,AREA,D	SHAPE	92
	\$ ,IH,L,MB,ML,MLC,MLW,NEW,NS,NST,IRITE)	SHAPE	93
C		SHAPE	94
	IF (ML(L).GT.MB) GO TO 825	SHAPE	95
C		SHAPE	96
	JMAX=0	SHAPE	97
	JHAK=0	SHAPE	98
	DO 570 I=1,L	SHAPE	99
	JHAK=MAX0(JHAK,MLC(2,I))	SHAPE	100
	570 JMAX=MAX0(JMAX,ML(I))	SHAPE	101
	JMX=JHAK	SHAPE	102
C		SHAPE	103
C	FIND ORDER OF LEADING EDGE BOX AT J-TH CHORDWISE ROW	SHAPE	104
C		SHAPE	105
	DO 620 J=1,JMAX	SHAPE	106
	EDG(J)=0.0	SHAPE	107
	DO 620 K=1,2	SHAPE	108
	620 MLT(K,J)=0	SHAPE	109
	I1=1	SHAPE	110
	I2=1	SHAPE	111
	K1=0	SHAPE	112
	DO 655 J=1,JMX	SHAPE	113
	630 IF (MLC(2,I1).GE.J) GO TO 632	SHAPE	114
	I1=I1+1	SHAPE	115
	GO TO 630	SHAPE	116
	632 NZ1=11	SHAPE	117
C	FIND ORDER OF TRAILING EDGE BOX AT J-TH CHORDWISE ROW	SHAPE	118
	634 IF(I2.GT.L) GO TO 638	SHAPE	119
	IF (MLC(2,I2).GT.0) K1=1	SHAPE	120
	IF(K1.EQ.0) GO TO 635	SHAPE	121
	IF (MLC(2,I2).EQ.0.AND.NZ1.NE.0) GO TO 646	SHAPE	122
	635 MLS=1ABS(MLW(I2))	SHAPE	123
	IF (MLW(I2).NE.0) GO TO 636	SHAPE	124
	I2=I2+1	SHAPE	125
	K2=1	SHAPE	126
	GO TO 634	SHAPE	127
	636 IF(MLS-J) 644,646,638	SHAPE	128
	638 IF(K2.LT.0) GO TO 648	SHAPE	129
	IF (MLW(I2)) 642,642,646	SHAPE	130
	642 I2=I2+K2	SHAPE	131
	IF(I2.LE.L) GO TO 634	SHAPE	132
	NZ2=L	SHAPE	133
	I2=L	SHAPE	134
	GO TO 652	SHAPE	135
	644 K2=MLW(I2)/MLS	SHAPE	136

GO TO 642	SHAPE	137
646 NZ2=I2-1	SHAPE	138
GO TO 652	SHAPE	139
648 NZ2=I2	SHAPE	140
652 CONTINUE	SHAPE	141
MLT(1,J)=NZ1	SHAPE	142
MLT(2,J)=NZ2	SHAPE	143
655 CONTINUE	SHAPE	144
C	SHAPE	145
C FIND ORDER OF WING BOXES AT EACH I-TH SPANWISE COLUMN	SHAPE	146
DO 670 I=1,L	SHAPE	147
IF (MLW(I)) 664,662,662	SHAPE	148
662 MLC(1,I)=MLW(I)+1	SHAPE	149
IF (MLC(2,I).EQ.0) MLC(1,I)=0	SHAPE	150
GO TO 670	SHAPE	151
664 MLC(2,I)=IABS(MLW(I))-1	SHAPE	152
MLC(1,I)=1	SHAPE	153
670 CONTINUE	SHAPE	154
C	SHAPE	155
IF (IRITE.EQ.0) GO TO 678	SHAPE	156
WRITE(IW,65)	SHAPE	157
IF (NEW.EQ.1) WRITE(IW,77)	SHAPE	158
IF (NEW.EQ.2) WRITE(IW,78)	SHAPE	159
WRITE(IW,82)	SHAPE	160
DO 672 J=1,JMX	SHAPE	161
672 WRITE(IW,85) J,MLT(1,J),MLT(2,J)	SHAPE	162
WRITE(IW,83)	SHAPE	163
DO 674 J=1,L	SHAPE	164
C 674 WRITE(IW,85) J,MLC(1,J),MLC(2,J)	SHAPE	165
674 WRITE(IW,85) J,MLC(1,J),MLC(2,J),MLW(J),ML(J)	SHAPE	166
678 CONTINUE	SHAPE	167
C	SHAPE	168
IF (XEDG(NSP).GT.1.) GO TO 865	SHAPE	169
NSP=NSP-1	SHAPE	170
DY=YEDG(NSP)-YEDG(NSP)	SHAPE	171
IF (IABS(DY).LE.LRRK) IEDG=1	SHAPE	172
IF (DY.LT.0.) GO TO 827	SHAPE	173
C COMPUTE VALUES FOR LEADING EDGE CORRECTION	SHAPE	174
YMAX2=YMAX*YMAX	SHAPE	175
DY=0	SHAPE	176
DO 720 J=1,JMX	SHAPE	177
IF (IEDG) 710,715,715	SHAPE	178
C DY IN FOLLOWING EXPRESSION IS ARBITRARY	SHAPE	179
710 EDG(J)=SQRT((YMAX2-YY(J)*YY(J))/(DY*(2.*YMAX-DY)))	SHAPE	180
IF (EDG(J).GT.1.0) GO TO 715	SHAPE	181
GO TO 720	SHAPE	182
715 EDG(J)=1.0	SHAPE	183
720 CONTINUE	SHAPE	184
C	SHAPE	185
750 CONTINUE	SHAPE	186
C	SHAPE	187
790 CONTINUE	SHAPE	188
C	SHAPE	189
RETURN	SHAPE	190
C	SHAPE	191
800 IERROR=800	SHAPE	192
T(1,1)=XEDG(K)	SHAPE	193
T(2,1)=YEDG(K)	SHAPE	194
IER=2	SHAPE	195
GO TO 843	SHAPE	196
805 IERROR=805	SHAPE	197
GO TO 840	SHAPE	198
810 IERROR=810	SHAPE	199
T(1,1)=DA(27)	SHAPE	200
T(2,1)=NSP	SHAPE	201
IER=2	SHAPE	202
GO TO 840	SHAPE	203
815 IERROR=815	SHAPE	204
GO TO 840	SHAPE	205
820 IERROR=820	SHAPE	206
T(1,1)=XEDG(NSP)	SHAPE	207

T(2,1)=YEDG(NSP)	SHAPE	208
T(3,1)=NSP	SHAPE	209
IER=3	SHAPE	210
GO TO 840	SHAPE	211
825 IERROR=825	SHAPE	212
T(1,1)=NEH	SHAPE	213
T(2,1)=K	SHAPE	214
T(3,1)=ML(K)	SHAPE	215
T(4,1)=ML(K-1)	SHAPE	216
IER=4	SHAPE	217
GO TO 840	SHAPE	218
827 IERROR=827	SHAPE	219
T(1,1)=YEDG(NSP)	SHAPE	220
T(2,1)=YEDG(NSM)	SHAPE	221
T(3,1)=NSP	SHAPE	222
T(4,1)=NSM	SHAPE	223
IER=4	SHAPE	224
GO TO 840	SHAPE	225
830 IERROR=730	SHAPE	226
T(1,1)=I	SHAPE	227
T(2,1)=J	SHAPE	228
T(3,1)=X	SHAPE	229
T(4,1)=Y	SHAPE	230
T(5,1)=EM	SHAPE	231
T(6,1)=EMY	SHAPE	232
T(7,1)=DEL	SHAPE	233
IER=7	SHAPE	234
840 WRITE(10,20) IERROR, (T(I,1),I=1,IER)	SHAPE	235
STOP	SHAPE	236
850 IPR=29	SHAPE	237
GO TO 890	SHAPE	238
855 IPR=24	SHAPE	239
GO TO 890	SHAPE	240
860 IPR=30	SHAPE	241
GO TO 890	SHAPE	242
865 K= MIN0(K,NS)	SHAPE	243
IPR=2*K+29	SHAPE	244
GO TO 890	SHAPE	245
871 IPR=45	SHAPE	246
890 WRITE	SHAPE	247
0 (10,10)IPR	SHAPE	248
10 FORMAT(1H0,10X,17HSHAPE -- BAD DATA,I5)	SHAPE	249
20 FORMAT(1H0,10X,38HBAD NUMBER IN SHAPE NEAR STATEMENT NO.,I5,	SHAPE	250
1 /1H,15X,1P8E14.6)	SHAPE	251
40 FORMAT(1H0,5X,3HNO.,12,43H REDISTRIBUTION OF WING LEADING EDGES,	SHAPE	252
1 NS(I,1,4H) = ,12,7H, 1EDG,3H = ,11)	SHAPE	253
45 FORMAT(1H0,5X,25HWING TRAILING EDGES, NST(,11,4H) = ,12,	SHAPE	254
5 10H, 1EDG = ,11)	SHAPE	255
50 FORMAT(4(6X,13,1P2E11.4))	SHAPE	256
60 FORMAT(1H1,5X,47HLOCAL MACH NUMBER DISTRIBUTION ON PHYSICAL WING/)	SHAPE	257
65 FORMAT(1H1)	SHAPE	258
70 FORMAT(1H0,5X,12,19H--TH SPANWISE COLUMN)	SHAPE	259
75 FORMAT(1H0,5X,12,17H--TH CHORDWISE ROW)	SHAPE	260
77 FORMAT(40X,13HPHYSICAL WING///)	SHAPE	261
78 FORMAT(40X,16HTRANSFORMED WING///)	SHAPE	262
80 FORMAT(1H,5X,6(2X,13,1P2E13.5))	SHAPE	263
82 FORMAT(15X,68HORDER OF FIRST(LEADING) AND LAST(TRAILING) WING BOX	SHAPE	264
11N CHORDWISE ROW//20X,1HJ,3X,12HMLT(NEW,1,J),3X,12HMLT(NEW,2,J)/)	SHAPE	265
83 FORMAT(///15X,62HORDER OF FIRST(ROOT) AND LAST(TIP) WING BOX. IN SP	SHAPE	266
1ANWISE COLUMN//20X,1HI,3X,12HMLC(NEW,1,I),3X,12HMLC(NEW,2,I)	SHAPE	267
2 ,4X,11H(MLW(NEW,1),5X,10HML(NEW,1))/)	SHAPE	268
85 FORMAT(16X,15,5X,15,3(10X,15))	SHAPE	269
86 FORMAT(20X,58HGAUSSIAN INTEGRATION POINTS IN CHORDWISE ROW - GX(NE	SHAPE	270
1W,K,J)/)	SHAPE	271
87 FORMAT(///20X,52HGAUSSIAN INTEGRATION POINTS IN SEMI-SPAN - GY(NEW	SHAPE	272
1W,K,J)/)	SHAPE	273
88 FORMAT(///20X,60HSPANWISE COORDINATE (GYP) AND MACH NUMBER (GMP) A	SHAPE	274
ST GX(2,K,J)/)	SHAPE	275
89 FORMAT(1H,5X,3(2X,13,1P2E16.0))	SHAPE	276
STOP	SHAPE	277
END	SHAPE	278

SUBROUTINE PLNFH(XEDG, YEDG, XTDC, YTDG, XX, YY, XLE, XTE, AR, AREA, D	PLNFH	2
IN, L, MB, HL, HLC, HLW, HEN, HS, NST, JRYTE)	PLNFH	3
DIMENSION XEDG(12), YEDG(12), XTDC(12), YTDG(12), XX(12), YY(12), AR(MB,1)	PLNFH	4
HLW(12), HLC(12,1), HLW(12), XLE(12), XTE(12)	PLNFH	5
DATA ERRR/1.E+06/, ERR1/0.001/, ERR2/1.E+05/, ANAK/0.5/	PLNFH	6
DATA Z1/ZHF=1, Z2/ZHG=1	PLNFH	7
DH=0.5*D	PLNFH	8
D2=0*D	PLNFH	9
NSTP=NST	PLNFH	10
DO 30 I=1, MB	PLNFH	11
HLC(2, I)=0	PLNFH	12
HLW(1)=0	PLNFH	13
XLE(1)=XEDG(NSTP)	PLNFH	14
XTE(1)=XEDG(NSTP)	PLNFH	15
DO 30 J=1, MB	PLNFH	16
30 AR(J, 1)=0.0	PLNFH	17
C	PLNFH	18
210 K1=0	PLNFH	19
K2=NST	PLNFH	20
JK=1	PLNFH	21
XR=XEDG(1)	PLNFH	22
YR=YEDG(1)	PLNFH	23
X2=XR	PLNFH	24
Y2=YR	PLNFH	25
XE2=0.0	PLNFH	26
SUM=0.0	PLNFH	27
AREA=0.0	PLNFH	28
ASNG=1.0	PLNFH	29
JLT=1	PLNFH	30
C	PLNFH	31
220 DO 530 I=1, L	PLNFH	32
ICHQ=0	PLNFH	33
IFIN=0	PLNFH	34
IEDZ=0	PLNFH	35
XL=XR	PLNFH	36
YL=YR	PLNFH	37
XR=XX(I)+DH	PLNFH	38
J=0	PLNFH	39
SAH=0.0	PLNFH	40
IF(ASNG) 225, 230, 230	PLNFH	41
225 IF(XR.LT.XTDC(1).AND.XTDC(1)-XR.GT.ERR2) GO TO 530	PLNFH	42
IF(K2.EQ.0) GO TO 440	PLNFH	43
GO TO 240	PLNFH	44
230 CONTINUE	PLNFH	45
IF (K1+1.GT.NS) GO TO 420	PLNFH	46
IF(K1.EQ.0) GO TO 400	PLNFH	47
240 ICHQ=0	PLNFH	48
IF(XR.GT.XE2) ICHQ=1	PLNFH	49
250 J=J+1	PLNFH	50
JFIN=0	PLNFH	51
Y=YY(J)+DH	PLNFH	52
IF(ASNG) 252, 256, 256	PLNFH	53
252 IF(ABS(XTDC(1)-XTDC(NSTP)).GT.ERR2.OR.I.NE.L) GO TO 254	PLNFH	54
IF(Y-DH.GT.YTDG(NSTP)) GO TO 530	PLNFH	55
XTE(J)=XTDC(1)	PLNFH	56
GO TO 250	PLNFH	57
254 SUM=SUM-AR(J, 1)	PLNFH	58
GO TO 258	PLNFH	59
256 AR(J, 1)=0.0	PLNFH	60
258 CONTINUE	PLNFH	61
IF(YL.LT.Y) GO TO 280	PLNFH	62
IF(ASNG) 265, 270, 270	PLNFH	63
265 AR(J, 1)=0.0	PLNFH	64
GO TO 250	PLNFH	65
270 CONTINUE	PLNFH	66
AR(J, 1)=1.0	PLNFH	67
SAH=SAH+AR(J, 1)	PLNFH	68
GO TO 250	PLNFH	69
	PLNFH	70

280	IF(Y-YL+ERR2.GE.D) GO TO 290	PLNFM	71
	AR(J,I)=AR(J,I)+ASNG*(YL-Y+D)/D	PLNFM	72
	X1=XL	PLNFM	73
	Y1=YL	PLNFM	74
	IF(ICHQ.NE.1) GO TO 310	PLNFM	75
	IF(JK) 284,.85,285	PLNFM	76
284	IF(YE2.LE.Y-D) GO TO 320	PLNFM	77
	GO TO 286	PLNFM	78
285	IF(YE2.GE.Y) GO TO 320	PLNFM	79
286	CONTINUE	PLNFM	80
	IF(XE1.GT.XL) GO TO 400	PLNFM	81
	IEDZ=2	PLNFM	82
	GO TO 400	PLNFM	83
290	X1=X2	PLNFM	84
	Y1=Y2	PLNFM	85
295	IEDZ=0	PLNFM	86
	IF(ICHQ.NE.1) GO TO 310	PLNFM	87
	YR=YE2	PLNFM	88
	GO TO 315	PLNFM	89
310	YR=YE1+TNG*(XR-XE1)	PLNFM	90
315	IF(JK) 317,316,316	PLNFM	91
316	IF(YR.GE.Y) GO TO 320	PLNFM	92
	GO TO 318	PLNFM	93
317	IF(YR.LE.Y-D) GO TO 320	PLNFM	94
318	CONTINUE	PLNFM	95
	IF(ICHQ.EQ.1) GO TO 325	PLNFM	96
	IFIN=1	PLNFM	97
	X2=XR	PLNFM	98
	Y2=YR	PLNFM	99
	GO TO 330	PLNFM	100
320	JFIN=1	PLNFM	101
	IF(JK) 322,321,321	PLNFM	102
321	Y2=Y	PLNFM	103
	GO TO 323	PLNFM	104
322	Y2=Y-D	PLNFM	105
323	X2=XE1+(Y2-YE1)/TNG	PLNFM	106
	IF(ABS(X2-XR).GT.ERRR) GO TO 330	PLNFM	107
	IFIN=1	PLNFM	108
	JFIN=0	PLNFM	109
	GO TO 330	PLNFM	110
325	IEDZ=1	PLNFM	111
	X2=XE2	PLNFM	112
	Y2=YE2	PLNFM	113
330	AR(J,I)=AR(J,I)+0.5*ASNG*(2.*XR-X2-X1)*(Y2-Y1)/D2	PLNFM	114
C		PLNFM	115
C		PLNFM	116
	IF(ABS(TNG).GT.ERRR) TNG=T-20	PLNFM	117
1331	CONTINUE	PLNFM	118
	IF(JK) 332,331,331	PLNFM	119
331	IF(J.LT.JLT) GO TO 334	PLNFM	120
	XLT=XE1+(YY(JLT)-YE1)/TNG	PLNFM	121
	IF(XLT.GT.XE2.AND.ASNG.GT.0.0) GO TO 334	PLNFM	122
	IF(YE2.GT.0.0.AND.YY(JLT).LE.YL.AND.ASNG.GT.0.0) XLT=0.0	PLNFM	123
	IF(YY(JLT).GE.YTGG(INSTP)) XLT=ATGG(INSTP)	PLNFM	124
	IF(ASNG.GT.0.0) XLT(JLT)=XLT	PLNFM	125
	IF(ASNG.LT.0.0) XLT(JLT)=XLT	PLNFM	126
	GO TO 333	PLNFM	127
332	IF(J.GT.JLT) GO TO 334	PLNFM	128
	XLT=XE1+(YY(JLT)-YE1)/TNG	PLNFM	129
	IF(YY(JLT).GE.YTGG(INSTP)) XLT=ATGG(INSTP)	PLNFM	130
	XTE(JLT)=XLT	PLNFM	131
333	JLT=JLT+JK	PLNFM	132
	GO TO 1331	PLNFM	133
334	CONTINUE	PLNFM	134
C		PLNFM	135
	IF(IEOZ.EQ.1) GO TO 400	PLNFM	136
	SAM=SAM+AS(J,I)	PLNFM	137
	IF(JK) 344,342,342	PLNFM	138
342	CONTINUE	PLNFM	139
	IF(JFIN.EQ.1) GO TO 200	PLNFM	140

GO TO 350	PLNFM	141
344 IF(JFIN.EQ.1) GO TO 475	PLNFM	142
350 IF(ASNG) 355,360,360	PLNFM	143
355 IF(IIFIN.EQ.1) GO TO 473	PLNFM	144
GO TO 530	PLNFM	145
360 IF(IIFIN.EQ.1) GO TO 465	PLNFM	146
GO TO 500	PLNFM	147
C	PLNFM	148
C LEADING EDGE AND TRAILING EDGE COMPUTATION	PLNFM	149
C	PLNFM	150
400 IF(ASNG) 440,465,405	PLNFM	151
405 IF(JK) 430,410,410	PLNFM	152
410 KI=KI+JK	PLNFM	153
IF(K1.GE.NSP) GO TO 420	PLNFM	154
XE1=XEDG(K1)	PLNFM	155
YE1=YEDG(K1)	PLNFM	156
K=KI+JK	PLNFM	157
XE2=XEDG(K)	PLNFM	158
YE2=YEDG(K)	PLNFM	159
GO TO 450	PLNFM	160
420 IF(JK.LT.0) GO TO 240	PLNFM	161
JK=-1	PLNFM	162
JLT=J	PLNFM	163
MLC(2,1)=0	PLNFM	164
K2=K2-JK	PLNFM	165
430 K2=K2+JK	PLNFM	166
XE1=XTDG(K2)	PLNFM	167
YE1=YTDG(K2)	PLNFM	168
K=K2+JK	PLNFM	169
XE2=XTDG(K)	PLNFM	170
YE2=YTDG(K)	PLNFM	171
GO TO 450	PLNFM	172
440 K2=K2+1	PLNFM	173
XE1=XTDG(K2)	PLNFM	174
YE1=YTDG(K2)	PLNFM	175
XE2=XTDG(K2+1)	PLNFM	176
YE2=YTDG(K2+1)	PLNFM	177
450 G=XE2-XE1	PLNFM	178
F=YE2-YE1	PLNFM	179
TNG=F/(G+1.E-20)	PLNFM	180
IF(G.LT.0.) GO TO 650	PLNFM	181
AREA=AREA+ASNG*G*(YE2+YE1)	PLNFM	182
C	PLNFM	183
ICHO=0	PLNFM	184
IF(XR.GT.XE2) ICHO=1	PLNFM	185
IF(IEDZ.EQ.1) GO TO 290	PLNFM	186
IF(IEDZ.EQ.2) GO TO 295	PLNFM	187
GO TO 250	PLNFM	188
C	PLNFM	189
C NUMBER OF BOXES IN SPANWISE COLUMN	PLNFM	190
C WAKE TYPE AND ORDER OF WAKE BOX	PLNFM	191
C	PLNFM	192
465 IF(ASNG) 478,467,467	PLNFM	193
467 IF(JK) 476,469,469	PLNFM	194
469 JJ=0	PLNFM	195
IF (AR(J,I).LT.ERR2) JJ=1	PLNFM	196
ML(I)=J-JJ	PLNFM	197
C	PLNFM	198
JJ=ML(I)	PLNFM	199
DO 471 K=1,JJ	PLNFM	200
KK=K	PLNFM	201
IF (AR(K,I).LT.AWAK) GO TO 472	PLNFM	202
471 CONTINUE	PLNFM	203
MLC(2,I)=KK	PLNFM	204
GO TO 473	PLNFM	205
472 MLC(2,I)=KK-1	PLNFM	206
473 CONTINUE	PLNFM	207
C	PLNFM	208
C ADJUSTMENT OF FIRST BOX AT LEADING EDGE ALONG KK-TH CHORDWISE ROW	PLNFM	209
KK=MLC(2,I)	PLNFM	209
IF (XX(I).GE.XLE(KK)) GO TO 474	PLNFM	210

MLC(2,1)=MLC(2,1)-1	PLNFM	211
GO TO 473	PLNFM	212
474 CONTINUE	PLNFM	213
C	PLNFM	214
GO TO 500	PLNFM	215
476 MLS=-1	PLNFM	216
ML(1)=ML(1-1)	PLNFM	217
K=ML(1)	PLNFM	218
GO TO 480	PLNFM	219
478 MLS=1	PLNFM	220
K=J	PLNFM	221
480 DO 490 JJ=1,K	PLNFM	222
IF (AR(JJ,1)) 481,484,484	PLNFM	223
481 IF (ABS(AR(JJ,1))-CHK2) 483,483,482	PLNFM	224
C	PLNFM	225
482 WRITE(IN,90) CHK2,JJ,1,AR(JJ,1)	PLNFM	226
C	PLNFM	227
483 AR(JJ,1)=0.0	PLNFM	228
484 CONTINUE	PLNFM	229
IF (MLW(1).NE.0) GO TO 490	PLNFM	230
IF (ASAC) 487,486,486	PLNFM	231
C	PLNFM	232
WIC TIP WAKE	PLNFM	233
486 IF (AR(JJ,1).GE.AWAK) GO TO 490	PLNFM	234
MLW(1)=JJ	PLNFM	235
GO TO 498	PLNFM	236
C	PLNFM	237
WING ROOT WAKE	PLNFM	238
487 IF (1-(JJ,1).LT.AWAK) GO TO 490	PLNFM	239
MLW(1)=JJ-1	PLNFM	240
488 MLW(1)=MLW(1)+PLS	PLNFM	241
489 CONTINUE	PLNFM	242
GO TO 500	PLNFM	243
C	PLNFM	244
TAILING EDGE BOX AREA ADJUSTMENT FOR WING TIP WAKE	PLNFM	245
C	PLNFM	246
490 CONTINUE	PLNFM	247
J=1	PLNFM	248
YES=0	PLNFM	249
IF IN=0	PLNFM	250
SAF=5-7-AR(JJ,1)	PLNFM	251
GO TO 490	PLNFM	252
C	PLNFM	253
END OF A SPANWISE COLUMN	PLNFM	254
C	PLNFM	255
500 CONTINUE	PLNFM	256
IF (ASAC) 517,517 GO TO 516	PLNFM	257
C	PLNFM	258
ADJUST WING TIP BOX FOR WING ROOT WAKE	PLNFM	259
GO=ML(1)	PLNFM	260
510 IF (1-(JJ,1).LT.AWAK) GO TO 512	PLNFM	261
MLC(1,1)=JJ	PLNFM	262
GO TO 515	PLNFM	263
512 JJ=JJ-1	PLNFM	264
IF (JJ) 514,514,510	PLNFM	265
514 MLC(2,1)=0	PLNFM	266
IF (MLW(1).EQ.0) GO TO 516	PLNFM	267
WRITE(IN,155) 1,1,MLW(1),1,MLC(2,1)	PLNFM	268
STOP	PLNFM	269
516 CONTINUE	PLNFM	270
CHK=0,MAXAM	PLNFM	271
C	PLNFM	272
530 CONTINUE	PLNFM	273
C	PLNFM	274
IF (ASAC) 700,700 GO TO 700	PLNFM	275
IF (XTDG(1).GE.XTDG(NSTP)) GO TO 700	PLNFM	276
C	PLNFM	277
INITIALIZATION FOR WING ROOT WAKE	PLNFM	278
C	PLNFM	279
K2=0	PLNFM	280
YE=0.0	PLNFM	
YR=0.0	PLNFM	
YEZ=XTDG(1)	PLNFM	

YE2=YTDG(1)	PLNFM	281
X2=XE2	PLNFM	282
Y2=YE2	PLNFM	283
ASNG=-1.0	PLNFM	284
JLT=1	PLNFM	285
GO TO 220	PLNFM	286
C ERROR MESSAGE	PLNFM	287
650 K=450	PLNFM	288
A=C	PLNFM	289
Z=Z2	PLNFM	290
WRITE(IW,150) K,Z,A	PLNFM	291
STOP	PLNFM	292
C	PLNFM	293
700 CONTINUE	PLNFM	294
IF(JRITE.EQ.0) GO TO 760	PLNFM	295
WRITE(IW,65)	PLNFM	296
IF(NEW.EQ.1) WRITE(IW,66)	PLNFM	297
IF(NEW.EQ.2) WRITE(IW,67)	PLNFM	298
DO 750 I=1,L	PLNFM	299
JL=ML(I)	PLNFM	300
K=JL/5	PLNFM	301
IF(JL.LE.5*K) GO TO 710	PLNFM	302
K=K+1	PLNFM	303
710 WRITE(IW,70) I	PLNFM	304
WRITE(IW,72) PLC(2,I)	PLNFM	305
IF (MLW(I)) 715,725,720	PLNFM	306
715 WRITE(IW,75) PLW(I)	PLNFM	307
GO TO 725	PLNFM	308
720 WRITE(IW,80) PLW(I)	PLNFM	309
725 DO 740 JJ=1,K	PLNFM	310
740 WRITE(IW,82) (J,AK(J,I),J=JJ,JL,K)	PLNFM	311
750 CONTINUE	PLNFM	312
C	PLNFM	313
JL=ML(L)	PLNFM	314
WRITE(6,140)	PLNFM	315
WRITE(IW,145) (J,XLE(J),XTE(J),J=1,JL)	PLNFM	316
C	PLNFM	317
SUM=2.*SUM*02	PLNFM	318
WRITE(IW,90) AREA,SUM	PLNFM	319
760 RETURN	PLNFM	320
65 FORMAT(1H,20X,37HNON-DIMENSIONAL BOX-AREA DISTRIBUTION)	PLNFM	321
66 FORMAT(1H,57X,21H - ( PHYSICAL WING)///)	PLNFM	322
67 FORMAT(1H,57X,21H - (TRANSFORMED WING)///)	PLNFM	323
70 FORMAT(//2X,12,14H-TM SPANWISE COLUMN//)	PLNFM	324
72 FORMAT(12X,16HFIRST L.C. BOX =,13,22H-TM BOX FROM WING ROOT)	PLNFM	325
75 FORMAT(12X,16HFIRST WAKE BOX =,13,22H-TM BOX FROM WING ROOT//)	PLNFM	326
80 FORMAT(12X,16HLAST WAKE BOX =,13,22H-TM BOX FROM WING ROOT//)	PLNFM	327
85 FORMAT(5X,5(1X,13,1PE11.4))	PLNFM	328
90 FORMAT(//10X,25HHING AREA CALCULATED FROM/	PLNFM	329
1 15X,26HLEADING AND TRAILING EDGES =,1PE11.4/	PLNFM	330
2 15X,26HSUMMATION OF AREA OF BOXES =,1PE11.4)	PLNFM	331
96 FORMAT(//10X,43HNEGATIVE BOX AREA EXCEEDS ALLOWABLE LIMIT (,	PLNFM	332
1 1PE11.4,1H)/10X,3HAR(,13,1H,,13,1H,,6HNEW) =,1PE11.4,	PLNFM	333
2 5X,21HCOMPUTATION CONTINUES//)	PLNFM	334
140 FORMAT(1H,9X,54HLEADING AND TRAILING EDGE POINTS AT EACH CHORDWIS	PLNFM	335
SE ROW/13X,1HJ,6X,7HLEADING,8X,8HTRAILING//)	PLNFM	336
145 FORMAT(10X,13,1P2E15.5)	PLNFM	337
150 FORMAT(1H0,4X,29HPLNFM--NEGATIVE VALUE NEAR SN,14,2X,A2,1PE11.4)	PLNFM	338
155 FORMAT(1H0,4X,25HORDER OF WING TIP BOX IN ,12,43H-TM SPANWISE COLU	PLNFM	339
IMN IS NOT PROPERLY DEFINED//	PLNFM	340
2 15X,8HMLW(NEW,,12,3H) =,13/15X,8HMLL(NEW,,12,3H) =,13)	PLNFM	341
END	PLNFM	342



	SUBROUTINE POT2(M2,M0,M0,CK,D,A)	POT2	2
C		POT2	3
C	THE VELOCITY FIELD OF A UNIFORM DOUBLET DISTRIBUTION	POT2	4
C	OVER A BOX IS COMPUTED AT ALL POINTS AT WHICH IT WILL BE	POT2	5
C	NEEDED AND STORED IN THE ARRAY A IN COMMON	POT2	6
C		POT2	7
C	M0,M0 CONTROL THE NUMBER OF VALUES COMPUTED	POT2	8
C		POT2	9
C	M2 IS THE RANGE OF THE SECOND SUBSCRIPT IN THE ARRAY,	POT2	10
C	DIMENSIONED A(2,M2,M2), BUT TREATED HERE AS AN ARRAY	POT2	11
C	WITH TWO SUBSCRIPTS	POT2	12
C		POT2	13
	DIMENSION A(2,1)	POT2	14
	M=M0	POT2	15
	N=N0	POT2	16
	DK=CK*D	POT2	17
	DK2=DK**2	POT2	18
	M1=M-1	POT2	19
	DK8=DK2/8.0	POT2	20
	DK4=2.0*DK8	POT2	21
	DK12=DK2/12.0	POT2	22
	CM=0.5	POT2	23
	DM=DK*0.5	POT2	24
	DM=0.5*DM	POT2	25
	DD=2.0*DK	POT2	26
	DDM=DD	POT2	27
	C1=0.25*DK2	POT2	28
	B5=CK2/24.0	POT2	29
	DO 3 I=1,M	POT2	30
	B1=0.0	POT2	31
	B4=2.0/DM	POT2	32
	B2=35/84-DM	POT2	33
	B3=-0.5*B5	POT2	34
	B3=DM*0.4*B5	POT2	35
	B4=DM*B4	POT2	36
	DC4=2.0*DM	POT2	37
	CM=1.0	POT2	38
	K=1	POT2	39
	C3=0.0	POT2	40
	C4=0.0	POT2	41
	C7=0.0	POT2	42
	C8=0.0	POT2	43
	DO 2 J=1,N	POT2	44
	A1=DM/CM	POT2	45
	C1=CM* COS(A1)	POT2	46
	C2=-CM* SIN(A1)	POT2	47
	C5=CM*CIN(A1,C6)	POT2	48
	C6=-CM*C6	POT2	49
	C9=C1-C3	POT2	50
	C10=C2-C4	POT2	51
	C11=C5-C7	POT2	52
	C12=C6-C8	POT2	53
	A(1,K)=B3*C9-B4*C10-B5*C3-B1*C11-B2*C12	POT2	54
	A(2,K)=B4*C9+B3*C10-B5*C4+B2*C11-B1*C12	POT2	55
23	C3=C1	POT2	56
	C4=C2	POT2	57
	C7=C5	POT2	58
	C8=C6	POT2	59
	B1=B1-DD	POT2	60
	B3=B3-DD	POT2	61
	B4=B4-DD	POT2	62
	D4=D4+DD4	POT2	63
	CM=CM+2.0	POT2	64
2	K=K+M2	POT2	65
	CM=CM+1.0	POT2	66
	DM=DM+DDM	POT2	67
3	DDM=DDM+DD	POT2	68
	DO 5 IL=1,2	POT2	69
	K1=1	POT2	70

DO 5 J=1,N	POT2	71
DO 4 I=1,M1	POT2	72
K=K1+M-1	POT2	73
4 A(IL,K)=A(IL,K)-A(IL,K-1)	POT2	74
A(IL,K1)=2.0*A(IL,K1)	POT2	75
5 K1=K1+M2	POT2	76
CM=0.0	POT2	77
DH=0.0	POT2	78
DDH=DK	POT2	79
DO 12 I=1,M	POT2	80
C7=0.0	POT2	81
C8=0.0	POT2	82
C9=0.0	POT2	83
C10=0.0	POT2	84
P1=0.0	POT2	85
P2=0.0	POT2	86
CN=1.0	POT2	87
B6=0.5*DK12	POT2	88
K=1	POT2	89
DO 10 J=1,N	POT2	90
A1=CM/CN	POT2	91
A2=DH/CN	POT2	92
IF (A1-0.2) 7,7,8	POT2	93
7 B1=2.0-A1*273.0	POT2	94
B2=-DK/(6.0*CN)	POT2	95
GO TO 9	POT2	96
8 B3= SIN(A1)/A1	POT2	97
B1=2.0*B3	POT2	98
B2=(B3- COS(A1))/A2-DH/CN*B3	POT2	99
9 B3= COS(A2)/CN	POT2	100
B4= SIN(A2)/CN	POT2	101
C3=B1*B3+B2*B4	POT2	102
C4=B2*B3-B1*B4	POT2	103
B5=DH*CN	POT2	104
C1=B5*C4-2.0*C3	POT2	105
C2=-2.0*C4-B5*C3	POT2	106
C5=C1-C7	POT2	107
C6=C2-C8	POT2	108
P3=P2-B6*CN	POT2	109
P4=P3+2.0*DK12*(CN-1.0)	POT2	110
A(1,K)=A(1,K)+C5-P1+C6+P3*C3-P4*C9	POT2	111
A(2,K)=A(2,K)+C6+P1+C5+P3*C4-P4*C10	POT2	112
P1=P1+DH	POT2	113
P2=P2+CN*DK4	POT2	114
CN=CN+2.0	POT2	115
C7=C1	POT2	116
C8=C2	POT2	117
C9=C3	POT2	118
C10=C4	POT2	119
B6=B6+DK12	POT2	120
10 K=K+M2	POT2	121
CM=CM+DK	POT2	122
DH=DH+DDH	POT2	123
12 DDH=DDH+DD	POT2	124
D3=CK/(2.0*3.14159265)	POT2	125
M1=M2-N	POT2	126
K=1	POT2	127
A1=0.0	POT2	128
DO 14 J=1,N	POT2	129
C1=D3* SIN(A1)	POT2	130
C2=-D3* COS(A1)	POT2	131
DO 13 I=1,M	POT2	132
DFF =A(1,K)*C1+A(2,K)*C2	POT2	133
A(2,K)=A(2,K)*C1-A(1,K)*C2	POT2	134
A(1,K)=DFF	POT2	135
13 K=K+1	POT2	136
K=K+M1	POT2	137
14 A1=A1+DH	POT2	138
RETURN	POT2	139
END	POT2	140

SUBROUTINE DRED(SFOX,SFDY,SFDH,KSF0,SFMX,SFMY,SFMH,KSPM,DA,T		DRED	2
,XX,YY,IH,L,MLC,M,MU,NB,NEW,NH,KSPS,IPRINT)		DRED	3
DIMENSION SFDX(NB,1),SFDY(NB,MU),SFDH(NH,MU),KSF0(1)		DRED	4
,SFMX(1),SFMY(1),SFMH(1),DA(1),T(NH,1),MLC(2,1)		DRED	5
,Z(1),YY(1)		DRED	6
C		DRED	7
ERRR=1.E-06		DRED	8
IRITE=IPRINT/10		DRED	9
JAITE=IPRINT/100		DRED	10
IF(NEW.LO.1) IRITE=IPRINT-10*IRITE		DRED	11
KSF0(M)=DA(98)		DRED	12
NP=KSP0(M)		DRED	13
IF (NP) 730,170,100		DRED	14
C		DRED	15
C	A SPLINE-SURFACE FOR THE DEFLECTION IS FITTED TO VALUES	DRED	16
C	OF DEFLECTION AT GIVEN POINTS.	DRED	17
C		DRED	18
100	IF (NB-NP) 730,120,120	DRED	19
120	CONTINUE	DRED	20
KP=100		DRED	21
DO 140 IP=1,NP		DRED	22
SFDX(IP,M)=DA(KP+1)/DA(24)		DRED	23
SFDY(IP,M)=DA(KP+2)/DA(24)		DRED	24
SFDH(IP,M)=DA(KP+3)		DRED	25
140	KP=KP+4	DRED	26
C	SPLINE-SURFACE FIT DATA	DRED	27
C		DRED	28
145	CONTINUE	DRED	29
IF(IRITE) 150,100,150		DRED	30
150	WRITE(IH,30) F	DRED	31
WRITE(IH,10)		DRED	32
C		DRED	33
100	CONTINUE	DRED	34
CALL SURFIT(NH,KSF0(M),T,SFDX(1,M),SFDY(1,M),SFDH(1,M),IRITE)		DRED	35
GO TO 200		DRED	36
C	PRESENTLY FOR PITCH AND PLUNGE OF FORM $Z=A0+A1*X$	DRED	37
170	SFDH(1,M)=DA(52)	DRED	38
SFDH(2,M)=DA(53)*DA(24)		DRED	39
SFDH(3,M)=0.0		DRED	40
GO TO 145		DRED	41
200	ICHECK=0	DRED	42
RETURN		DRED	43
730	IPR=98	DRED	44
GO TO 750		DRED	45
750	WRITE	DRED	46
(IH,20)IPR		DRED	47
STOP		DRED	48
10	FORMAT(IH0,10X,73HCOMPUTED DEFLECTION = A0+A1*X+A2*Y+ SUM OF H(I)*	DRED	49
1(R(I)**2)+(ALCG(R(I)**2)))		DRED	50
20	FORMAT(IH0,10X,16HRED -- BAD DATA,15)	DRED	51
30	FORMAT(IH0,8X,26HPHYSICAL PLANE -- MODE NO.,13)	DRED	52
END		DRED	53

	SUBROUTINE	HVAL(XX,YY,XTC,YTC,XTE,SFOX,SFOY,SFDH,KSFD,S,D,CR	HVAL	2
		,IW,E,HL,HI,MEN,HST,HB,HO,NB,NH,KSFS,IPRINT)	HVAL	3
C		CALCULATES DOWNWASH VELOCITY DISTRIBUTION (REAL AND IMAGINARY)	HVAL	4
		DIMENSION XX(1),YY(1),XTC(1),YTC(1),XTE(1),S(2,HB,1),HL(1)	HVAL	5
		,SFOX(HB,1),SFOY(HB,ND),SFDH(HH,MD),KSFD(1)	HVAL	6
		IRITE=IPRINT/10	HVAL	7
		IF(MEN.EQ.1) IRITE=IPRINT-10*IRITE	HVAL	8
		JLP=KSFD(H1)	HVAL	9
		IF(KSFD(H1).EQ.0.AND.MEN.EQ.2) JLP=KSFS	HVAL	10
		JL=HL(1)	HVAL	11
		DO 5 I=1,L	HVAL	12
		DO 5 J=1,JL	HVAL	13
		DO 5 K=1,2	HVAL	14
		5 S(K,J,I)=0.0	HVAL	15
		DO 80 I=1,L	HVAL	16
		JL=HL(I)	HVAL	17
		IF(JL) 40,80,40	HVAL	18
		40 CONTINUE	HVAL	19
		DO 70 J=1,JL	HVAL	20
		IF (XX(I).GT.(XTE(J)+0)) GO TO 70	HVAL	21
		CALL SURF2(XX(I),YY(J),1,1,1,VALU,VALUD,OH,SFOX(1,H1)	HVAL	22
		,SFOY(1,H1),SFDH(1,H1),JLP,2)	HVAL	23
		S(1,J,I)=VALUE	HVAL	24
		S(2,J,I)=CR*VALU	HVAL	25
		70 CONTINUE	HVAL	26
		80 CONTINUE	HVAL	27
		IF(IRITE) 100,200,100	HVAL	28
		100 WRITE (IW,10)	HVAL	29
		WRITE (IW,12) H1	HVAL	30
		10 FORMAT(1H,1CX,47HUPWASH (REAL, IMAGINARY, ABSOLUTE, PHASE ANGLE))	HVAL	31
		12 FORMAT(1H+,58X,28H---PHYSICAL PLANE---MODE NO.,13/)	HVAL	32
		DO 170 I=1,L	HVAL	33
		JL=HL(I)	HVAL	34
		IF (JL) 110,170,110	HVAL	35
		110 WRITE (IW,20) I	HVAL	36
		JLP=JL/2	HVAL	37
		IF (JL-2*JLP.NE.0) JLP=JLP+1	HVAL	38
		DO 160 J=1,JLP	HVAL	39
		S1=SQRT(S(1,J,I)*S(1,J,I)+S(2,J,I)*S(2,J,I))	HVAL	40
		IF(S1.GT.0.0) GO TO 120	HVAL	41
		S2=0.0	HVAL	42
		GO TO 130	HVAL	43
		120 S2=57.29578*ATAN2(S(2,J,I),S(1,J,I))	HVAL	44
		130 J1=J+JLP	HVAL	45
		IF (J1.LE.JL) GO TO 150	HVAL	46
		J1=0	HVAL	47
		S3=0.0	HVAL	48
		S4=0.0	HVAL	49
		GO TO 160	HVAL	50
		150 CONTINUE	HVAL	51
		S3=SQRT(S(1,J1,I)*S(1,J1,I)+S(2,J1,I)*S(2,J1,I))	HVAL	52
		IF(S3.GT.0.0) GO TO 150	HVAL	53
		S4=0.0	HVAL	54
		GO TO 160	HVAL	55
		155 S4=57.29578*ATAN2(S(2,J1,I),S(1,J1,I))	HVAL	56
		160 WRITE (IW,25) J,S(1,J,I),S(2,J,I),S1,S2,J1,S(1,J1,I),S(2,J1,I),S3,S4	HVAL	57
		170 CONTINUE	HVAL	58
C		THESE ARE THE UPWASHES	HVAL	59
C			HVAL	60
		200 CONTINUE	HVAL	61
		RETURN	HVAL	62
		20 FORMAT(1H0,5X,12,19H---TH SPANWISE COLUMN)	HVAL	63
		25 FORMAT(1H,5X,2(2X,13,1P4E13.5))	HVAL	64
		END	HVAL	65

	SUBROUTINE BOXP(XX,YY,XTDG,YTDG,XTE,XLE,A,AR,DAN,T,EDG,S,CK,D,YE	BOXP	2
1	IPRINT,MLT,MLC,AMA	BOXP	3
2	JMAX,L,ML,MLW,M1,NEW,NST,IW,MB,MC,ME,MD)	BOXP	4
C	ICHECK=0 - POTENTIAL AS COMPUTED IN BOXPO	BOXP	5
C	ICHECK=1 - WITH LEADING EDGE CORRECTION	BOXP	6
C	ICHECK=2 - PRESSURE COEFFICIENT COMPUTED FROM CORRECTED POTENTIAL	BOXP	7
C	ICHECK=3 - PRESSURE COEFFICIENT COMPUTED FROM TRANSFORMED VALUE	BOXP	8
C	FOR PHYSICAL WING	BOXP	9
	DIMENSION XX(1),YY(1),XTDG(1),YTDG(1),XTE(1),XLE(1),A(2,MC,MB)	BOXP	10
1	AM(MB,1),CDG(1),S(2,MB,MB),T(ME,ME,1)	BOXP	11
2	ML(1),MLW(1),MLT(2,1),MLC(2,MB),AMA(MB,MB)	BOXP	12
	IPRIN=IPRINT	BOXP	13
	KRITE=IPRIN/ICCOU	BOXP	14
	IPRIN=IPRIN-ICCOU*KRITE	BOXP	15
	JRITE=IPRIN/ICG	BOXP	16
	IPRIN=IPRIN-ICG*JRITE	BOXP	17
	IRITE=IPRIN	BOXP	18
	IF(NEW.EQ.1) GO TO 140	BOXP	19
	KRITE=KRITE/10	BOXP	20
	JRITE=JRITE/10	BOXP	21
	IRITE=IRITE/10	BOXP	22
	GO TO 150	BOXP	23
140	KRITE=KRITE-(KRITE/10)*10	BOXP	24
	JRITE=JRITE-(JRITE/10)*10	BOXP	25
	IRITE=IRITE-(IRITE/10)*10	BOXP	26
150	CONTINUE	BOXP	27
	NSMOS=0	BOXP	28
	DH=0.5*0	BOXP	29
C	CALL BOXPO(XX,YY,XTDG,YTDG,XTE,A,AR,DAN,T,S,CK,D,AMA	BOXP	30
3	IW,L,MB,MC,ML,MLC,MLW,M1,NEW,NST,JRITE)	BOXP	31
		BOXP	32
C	BOXPO COMPUTES THE POTENTIAL VALUES IN EACH BOX.	BOXP	33
C	THEY ARE STORED IN THE ARRAY'S.	BOXP	34
C		BOXP	35
	ICHECK=0	BOXP	36
	NEW=NEW	BOXP	37
	JMAX=JMAX	BOXP	38
	IF(IRITE.EQ.0) GO TO 270	BOXP	39
	WRITE(IW,30)	BOXP	40
200	DO 205 I=1,L	BOXP	41
	DO 205 J=1,JMAX	BOXP	42
	DO 205 K=1,2	BOXP	43
205	T(J,I,K)=S(K,J,I)	BOXP	44
215	WRITE(IW,32) P1	BOXP	45
30	FORMAT(IH1,10X,6HPOTENTIAL CALCULATED (REAL, IMAGINARY, ABSOLUTE,	BOXP	46
	\$ PHASE ANGLE))	BOXP	47
32	FORMAT(IH+,71X,28H---PHYSICAL PLANE---MODE NO.,13/)	BOXP	48
C	PRINT-OUT	BOXP	49
218	DO 250 I=1,L	BOXP	50
	JL=ML(1)	BOXP	51
	IF (JL) 220,250,220	BOXP	52
220	WRITE(IW,20) I	BOXP	53
	JLP=JL/2	BOXP	54
	IF(JL-2*JLP.NE.0) JLP=JLP+1	BOXP	55
	DO 240 J=1,JLP	BOXP	56
	S1=SQRT(T(J,I,1)*T(J,I,1)+T(J,I,2)*T(J,I,2))	BOXP	57
	IF(S1.NE.0.0) GO TO 224	BOXP	58
	S2=0.0	BOXP	59
	GO TO 226	BOXP	60
224	CONTINUE	BOXP	61
	S2=57.29578*ATAN2(T(J,I,2),T(J,I,1))	BOXP	62
226	CONTINUE	BOXP	63
	J1=J+JLP	BOXP	64
	IF(J1.LE.JL) GO TO 230	BOXP	65
	J1=0	BOXP	66
	S3=0.0	BOXP	67
	GO TO 230	BOXP	68
	T(J1,I,1)=0.0	BOXP	69
		BOXP	70

	T(J1,I,2)=0.0	BOXP	71
	GO TO 243	BOXP	72
230	CONTINUE	BOXP	73
	S3=500*(T(J1,I,1)+T(J1,I,1)+T(J1,I,2)+T(J1,I,2))	BOXP	74
	IF(S3.EQ.0) GO TO 234	BOXP	75
	S4=0.0	BOXP	76
	GO TO 236	BOXP	77
234	CONTINUE	BOXP	78
	S4=17.74-7.44*(T(J1,I,2)+T(J1,I,1))	BOXP	79
236	CONTINUE	BOXP	80
240	WRITE(IH,25) J,T(J,I,1),T(J,I,2),S1,S2	BOXP	81
	* J1,T(J1,I,1),T(J1,I,2),S3,S4	BOXP	82
250	CONTINUE	BOXP	83
C		BOXP	84
270	CONTINUE	BOXP	85
	IF(ICHECK.GE.1) GO TO 400	BOXP	86
C	LEADING EDGE CORRECTION	BOXP	87
	DO 300 J=1,JMAX	BOXP	88
	IF (MLT(1,J).EQ.0) GO TO 285	BOXP	89
	I1=MLT(1,J)	BOXP	90
	I2=MLT(2,J)	BOXP	91
	S3=0.0	BOXP	92
	DO 280 I=I1,I2	BOXP	93
	K=1	BOXP	94
	IF (AR(J,I).GE.1.0) GO TO 290	BOXP	95
280	CONTINUE	BOXP	96
285	S3=0	BOXP	97
290	IF(K.EQ.1) S3=0	BOXP	98
	S3=XX(K)-XLE(J)-DH+S3	BOXP	99
	DO 370 I=1,L	BOXP	100
	S1=(XX(I)-XLE(J))/S3	BOXP	101
	IF(S1) 300,340,310	BOXP	102
300	S1=0.0	BOXP	103
	GO TO 340	BOXP	104
310	IF(YE) 320,320,330	BOXP	105
320	S1=S1+(XX(I)+XLE(J))/(S3+2.*XLE(J))	BOXP	106
330	IF(S1.GT.1.0) S1=1.0	BOXP	107
	S1=EDG(J)+SQRT(S1)	BOXP	108
340	DO 350 K=1,2	BOXP	109
350	S(K,J,I)=S1*S(K,J,I)	BOXP	110
370	CONTINUE	BOXP	111
380	CONTINUE	BOXP	112
	ICHECK=1	BOXP	113
	IF(IRITE) 390,400,390	BOXP	114
390	WRITE(IH,36)	BOXP	115
36	FORMAT(IH1,10X,61HPOTENTIAL CORRECTED (REAL, IMAGINARY, ABSOLUTE,	BOXP	116
	\$ PHASE ANGLE))	BOXP	117
C	TRANSFER TO THE PRINT-OUT OF CORRECTED VELOCITY POTENTIAL	BOXP	118
	GO TO 200	BOXP	119
400	CONTINUE	BOXP	120
	IF(IRITE.EQ.0) GO TO 700	BOXP	121
	IF(ICHECK.GE.2) GO TO 520	BOXP	122
C	CALCULATE AND PRINT PRESSURE COEFFICIENT	BOXP	123
	ICHECK=2	BOXP	124
	DO 410 I=1,L	BOXP	125
	JL=ML(I)	BOXP	126
	DO 410 J=1,JL	BOXP	127
	T(J,I,3)=S(1,J,I)	BOXP	128
410	T(J,I,4)=S(2,J,I)	BOXP	129
420	JMAX=JMAX	BOXP	130
	DO 430 I=1,L	BOXP	131
	JL=ML(I)	BOXP	132
	DO 430 J=1,JL	BOXP	133
	DO 430 K=1,2	BOXP	134
430	T(J,I,K)=0.0	BOXP	135
	DO 500 J=1,JMAX	BOXP	136
	I1=MLT(1,J)	BOXP	137
	I2=MLT(2,J)-MLT(1,J)+1	BOXP	138
	I1=I1	BOXP	139
	IK=I2+1	BOXP	140

T(1, 6,5)=XLE(J)	BOXP	141
T(1, 8,5)=XLE(J)	BOXP	142
T(1,10,5)=0.0	BOXP	143
T(1,12,5)=0.0	BOXP	144
IF (ABS(XLE(J)-XX(I))) .GT. 1.E-05) GO TO 445	BOXP	145
C ADJUSTMENT -- LEADING EDGE AND FIRST BOX COINCIDES	BOXP	146
II=II+1	BOXP	147
IK=IK-1	BOXP	148
445 CONTINUE	BOXP	149
DO 450 I=2,IK	BOXP	150
T(I, 6,5)=XX(I)	BOXP	151
T(I, 8,5)=XX(I)	BOXP	152
T(I,10,5)=T(J,II,3)	BOXP	153
T(I,12,5)=T(J,II,4)	BOXP	154
T(I-T(I,1,5)=XX(II)	BOXP	155
450 II=II+1	BOXP	156
CALL SPLNI(I2,T(1, 1,5),T(1, 2,5),T(1, 4,5)	BOXP	157
,IK,T(1, 6,5),T(1,10,5),T(1,14,5),NSMOS)	BOXP	158
CALL SPLNI(I2,T(1, 1,5),T(1, 3,5),T(1, 5,5)	BOXP	159
,IK,T(1, 8,5),T(1,12,5),T(1,14,5),NSMOS)	BOXP	160
II=II	BOXP	161
DO 470 I=1,I2	BOXP	162
T(J,II,1)=T(I,4,5)-CK*T(I,3,5)	BOXP	163
T(J,II,2)=T(I,5,5)+CK*T(I,2,5)	BOXP	164
470 II=II+1	BOXP	165
500 CONTINUE	BOXP	166
WRITE(IH,40)	BOXP	167
40 FORMAT(1H1,10X,61HPRESSURE COEFFICIENT (REAL, IMAGINARY, ABSOLUTE,	BOXP	168
1 PHASE ANGLE))	BOXP	169
C TRANSFER TO PRINT-OUT SECTION	BOXP	170
GO TO 215	BOXP	171
520 CONTINUE	BOXP	172
700 CONTINUE	BOXP	173
RETURN	BOXP	174
20 FORMAT(1H0,5X,I2,19H--TH SPANWISE COLUMN)	BOXP	175
25 FORMAT(1H ,5X,2(2X,I3,1P4E13.5))	BOXP	176
END	BOXP	177

	SUBROUTINE BOXPO(XX,YY,XTOG,YTOG,XTE,A,AR,DAN,T,S,CK,D,AMA	BOXPO	2
	,IM,L,MB,MC,ML,MLC,MLH,MI,NEH,NST,JRITE)	BOXPO	3
C	SOLUTION OF SIMULTANEOUS EQUATIONS FOR THE POTENTIAL	BOXPO	4
	DIMENSION XX(1),YY(1),XTOG(1),YTOG(1),XTE(1),A(2,MC,1)	BOXPO	5
1	,AR(MB,MB),T(2,MB,1),S(2,MB,1),AMA(MB,MB)	BOXPO	6
2	,ML(1),MLC(2,1),MLH(1)	BOXPO	7
	CH=0.5*D	BOXPO	8
	DD=2.0*D	BOXPO	9
	LI=2+2*ML(L)	BOXPO	10
	IF(JRITE.EQ.0) GO TO 25	BOXPO	11
	IF(NEH.EQ.2.OR.MI.GT.1) GO TO 25	BOXPO	12
C	PRINT INFLUENCE COEFFICIENT	BOXPO	13
	WRITE(IM,100)	BOXPO	14
	JL=2*ML(L)	BOXPO	15
	K=JL/4	BOXPO	16
	IF(JL-4*K.NE.0) K=K+1	BOXPO	17
	DO 20 I=1,L	BOXPO	18
	I1=I-1	BOXPO	19
	WRITE(IM,110) I1	BOXPO	20
	DO 15 JI=1,K	BOXPO	21
15	WRITE(IM,120) ((J,A(1,J,I),A(2,J,I)),J=J1,JL,K)	BOXPO	22
20	CONTINUE	BOXPO	23
25	CONTINUE	BOXPO	24
	I1=NST	BOXPO	25
	AMI=1.0	BOXPO	26
	DO 26 I=1,I1	BOXPO	27
26	AMI=AMINI(XTOG(I),AMI)	BOXPO	28
C		BOXPO	29
	IF(JRITE.EQ.0) GO TO 28	BOXPO	30
	WRITE(IM,160)	BOXPO	31
28	CONTINUE	BOXPO	32
	I1=0	BOXPO	33
	MFLNS = DAN	BOXPO	34
	DO 90 I=1,L	BOXPO	35
	X=XX(I)	BOXPO	36
C	ADJUST UPSTREAM INFLUENCE	BOXPO	37
	K0 = 1	BOXPO	38
	IF(MFLNS.EQ.0) GO TO 30	BOXPO	39
	K0 = MAX(1,I-MFLNS+1)	BOXPO	40
30	CONTINUE	BOXPO	41
	JL=ML(I)	BOXPO	42
	IF (JL.EQ.0) GO TO 90	BOXPO	43
C	DEFINE WING AND WAKE BOXES	BOXPO	44
C		BOXPO	45
C		BOXPO	46
C		BOXPO	47
C		BOXPO	48
C		BOXPO	49
C		BOXPO	50
	JW=0	BOXPO	51
	IF(X.LE.AMI) GO TO 34	BOXPO	52
	J=1	BOXPO	53
32	IF(J.GT.JL) GO TO 34	BOXPO	54
	IF (X.LE.(XTE(J)+DH)) GO TO 33	BOXPO	55
	JW=JW+1	BOXPO	56
33	J=J+1	BOXPO	57
	GO TO 32	BOXPO	58
34	JE=JL-JW	BOXPO	59
	IF (MLH(I)) 36,35,35	BOXPO	60
35	JS = JW+1	BOXPO	61
	JSW=1	BOXPO	62
	GO TO 37	BOXPO	63
36	JS = 1	BOXPO	64
	JSW=JE +1	BOXPO	65
37	JN = JS +JE-1	BOXPO	66
	JNW=JSW+JW-1	BOXPO	67
	IF (I1.EQ.0) GO TO 50	BOXPO	68
C	SUBTRACT ACTION OF CONTRIBUTIONS OF PRECEDING ROWS TO UPWASH	BOXPO	69
	GO 47 J=JS,JN	BOXPO	70



DO 45 K=KQ,I1	BOXPO	71
KL=ML(K)	BOXPO	72
K1=I+1-K	BOXPO	73
IF (KL.EQ.0) GO TO 45	BOXPO	74
DO 40 N=1,K'	BOXPO	75
IF (NEW.EQ.2) GO TO 38	BOXPO	76
N1=N+J	BOXPO	77
N2=IABS(N-J)+1	BOXPO	78
A1=A(1,N1,K1)+A(1,N2,K1)	BOXPO	79
A2=A(2,N1,K1)+A(2,N2,K1)	BOXPO	80
GO TO 39	BOXPO	81
38 CONTINUE	BOXPO	82
TEMP=(AMA(J,I)+AMA(N,K))/DD	BOXPO	83
YM1=(YY(J)+YY(N))*TEMP	BOXPO	84
YM2=(YY(J)-YY(N))*TEMP	BOXPO	85
YM2=ABS(YM2)	BOXPO	86
N1=YM1	BOXPO	87
N2=YM2	BOXPO	88
YM1=YM1-FLOAT(N1)	BOXPO	89
YM2=YM2-FLOAT(N2)	BOXPO	90
N1=N1+1	BOXPO	91
N2=N2+1	BOXPO	92
IF (N1.GT.LIM2) GO TO 138	BOXPO	93
ARN10=A(1,N1,K1)	BOXPO	94
AIN10=A(2,N1,K1)	BOXPO	95
IF (N1+1.GT.LIM2) GO TO 238	BOXPO	96
ARN11=A(1,N1+1,K1)	BOXPO	97
AIN11=A(2,N1+1,K1)	BOXPO	98
GO TO 338	BOXPO	99
138 ARN10=A(1,LIM2,K1)	BOXPO	100
AIN10=A(2,LIM2,K1)	BOXPO	101
238 ARN11=A(1,LIM2,K1)	BOXPO	102
AIN11=A(2,LIM2,K1)	BOXPO	103
338 CONTINUE	BOXPO	104
ARN20=A(1,N2,K1)	BOXPO	105
AIN20=A(2,N2,K1)	BOXPO	106
ARN21=A(1,N2+1,K1)	BOXPO	107
AIN21=A(2,N2+1,K1)	BOXPO	108
A1=ARN10+(ARN11-ARN10)*YM1+ARN20+(ARN21-ARN20)*YM2	BOXPO	109
A2=AIN10+(AIN11-AIN10)*YM1+AIN20+(AIN21-AIN20)*YM2	BOXPO	110
A1=A1*AMA(N,K)	BOXPO	111
A2=A2*AMA(N,K)	BOXPO	112
39 CONTINUE	BOXPO	113
WT=1.0	BOXPO	114
IF (N.GE.MLC(2,K).AND.MLC(2,K).NE.0) WT=AR(N,K)	BOXPO	115
S(1,J,I)=S(1,J,I)-(A1*S(1,N,K)-A2*S(2,N,K))*WT	BOXPO	116
40 S(2,J,I)=S(2,J,I)-(A2*S(1,N,K)+A1*S(2,N,K))*WT	BOXPO	117
45 CONTINUE	BOXPO	118
47 CONTINUE	BOXPO	119
C SETTING UP MATRIX FOR SIMULTANEOUS EQUATIONS	BOXPO	120
50 DO 52 J=1,JL	BOXPO	121
DO 52 K=1,JL	BOXPO	122
N1=J+K	BOXPO	123
N2=IABS(J-K)+1	BOXPO	124
WT=1.0	BOXPO	125
IF (K.GE.MLC(2,I).AND.MLC(2,I).NE.0) WT=AR(K,I)	BOXPO	126
C IF (J.EQ.K) WT=1.0	BOXPO	127
C	BOXPO	128
C	BOXPO	129
T(1,J,K)=(A(1,N1,I)+A(1,N2,I))*WT	BOXPO	130
T(2,J,K)=(A(2,N1,I)+A(2,N2,I))*WT	BOXPO	131
52 CONTINUE	BOXPO	132
C SUBTRACTION OF CONTRIBUTION	BOXPO	133
C FROM WAKE BOXES -- S(WING)=	BOXPO	134
C S(WING)-T(WAKE)*PHI(WAKE)	BOXPO	135
IF (JW.EQ.0) GO TO 60	BOXPO	136
DO 56 J=JS,JN	BOXPO	137
DO 55 N=JSH,JNH	BOXPO	138
S(1,J,I)=S(1,J,I)-T(1,J,N)*S(1,N,I)+T(2,J,N)*S(2,N,I)	BOXPO	139
55 S(2,J,I)=S(2,J,I)-T(1,J,N)*S(2,N,I)+T(2,J,N)*S(1,N,I)	BOXPO	140

56	CONTINUE	BOXPO	141
C	IF (MLW(1),LT.0) GO TO 60	BOXPO	142
	RE-POSITION ELEMENTS OF Y	BOXPO	143
	DO 59 N=1,JE	BOXPO	144
	NW=N+JW	BOXPO	145
	DO 58 J=1,JE	BOXPO	146
	NH=J+JW	BOXPO	147
	DO 58 K=1,2	BOXPO	148
	Y(K,J,N)=Y(K,NH,NH)	BOXPO	149
	59 CONTINUE	BOXPO	150
C		BOXPO	151
C	SOLUTION OF EQUATIONS	BOXPO	152
C	60 CONTINUE	BOXPO	153
	IF (JRITE.EQ.0) GO TO 72	BOXPO	154
	WRITE(IH,167) I	BOXPO	155
	DO 70 K=1,2	BOXPO	156
	N1=1	BOXPO	157
	N2=N1+3	BOXPO	158
	DO 67 J=1,JE	BOXPO	159
	67 WRITE(IW,169) (J,N,Y(K,J,N),N=N1,N2)	BOXPO	160
	IF (N2.GE.JE) GO TO 63	BOXPO	161
	N1=N2+1	BOXPO	162
	N2=N1+3	BOXPO	163
	GO TO 66	BOXPO	164
	68 CONTINUE	BOXPO	165
	DO 69 J=1,JE	BOXPO	166
	WRITE(IW,170) K,J,1,Y(K,J,1)	BOXPO	167
	69 CONTINUE	BOXPO	168
	70 CONTINUE	BOXPO	169
	167 FORMAT(IH0,5X,I2,1,M-TH SPANWISE COLUMN)	BOXPO	170
	168 FORMAT(IH1,10X,64HCOEFFICIENT MATRIX (A) OF (A)*(X)=(B) FOR VELO	BOXPO	171
	SCITY POTENTIAL (X) ALONG EACH COLUMN//)	BOXPO	172
	169 FORMAT(8X,4(2X,1H,12,1H,12,2H),E13.6)	BOXPO	173
	170 FORMAT(9X,2HS(,12,1H,12,1H,12,2H)=,E13.6)	BOXPO	174
	72 CONTINUE	BOXPO	175
C		BOXPO	176
	K = MSIMEC(MB,JE,1,T,S(1,JS,I))	BOXPO	177
	IF (K.NE.1) GO TO 90	BOXPO	178
C		BOXPO	179
C	COMPUTE WAKE POTENTIALS --	BOXPO	180
	PHIITE)*EXP(-IK*(X-XTE))	BOXPO	181
	IF (X.LL.(AMI-C)) GO TO 85	BOXPO	182
	JS=1	BOXPO	183
75	Y=YY(JS)	BOXPO	184
	IF (X.LT.(XTE(JS)-DH)) GO TO 82	BOXPO	185
	IF (X.GE.(XTE(JS)+DH)) GO TO 82	BOXPO	186
	IF (X.GT.XTE(JS)) GO TO 76	BOXPO	187
	PTR=S(1,JS,I)	BOXPO	188
	PTI=S(2,JS,I)	BOXPO	189
	KK=1	BOXPO	190
	XB=X+D	BOXPO	191
	GO TO 77	BOXPO	192
76	P=(X-XTE(JS))/C	BOXPO	193
	PTR=(1.0-P)*S(1,JS,I)+P*S(1,JS,I-1)	BOXPO	194
	PTI=(1.0-P)*S(2,JS,I)+P*S(2,JS,I-1)	BOXPO	195
	KK=0	BOXPO	196
	XB=X	BOXPO	197
77	CONTINUE	BOXPO	198
80	IF (XB.GT.1.0) GO TO 82	BOXPO	199
	XW=(XB-XTE(JS))*CK	BOXPO	200
	IKW=I+KK	BOXPO	201
	S(1,JS,IKW)=PTR*COS(XW)+PTI*SIN(XW)	BOXPO	202
	S(2,JS,IKW)=PTI*COS(XW)-PTR*SIN(XW)	BOXPO	203
	KK=KK+1	BOXPO	204
	XB=XB+D	BOXPO	205
	GO TO 80	BOXPO	206
82	JS=JS+1	BOXPO	207
	IF (JS.LE.JL) GO TO 75	BOXPO	208
85	CONTINUE	BOXPO	209
90	II=II+1	BOXPO	210

RETURN	80XPO	211
95 WRITE (17,140)	80XPO	212
STOP	80XPO	213
100 FORMAT(1H1,20X-42HINFLUENCE COEFFICIENT (REAL AND IMAGINARY)///)	80XPO	214
110 FORMAT(1H0,3X,13,38H-BOX SEPARATION IN CHORDWISE DIRECTION)	80XPO	215
120 FORMAT(1H,5X,4(1X,1P2E13.6))	80XPO	216
140 FORMAT(1H010X,59HSOLUTION OF SIMULTANEOUS EQUATIONS FOR THE POTENT	80XPO	217
IAL FAILED)	80XPO	218
END	80XPO	219

	SUBROUTINE FORCI(XX,YY,S,SFDX,SFDY,SFDH,KSFH,KSFY,XLE,XTE,YMAX	FORCI	2
1	,AMA,JHAR,MLT,L,NEW,MZ,MB,MD,NB,NH,CK,T)	FORCI	3
	DIMENSION MLT(2,MB),XLE(MB),XTE(MB),XX(MB),YY(MB),S(2,MB,MB),	FORCI	4
1	,SFDX(NB,MD),SFDY(NB,MD),SFDH(NH,MD),KSFH(MD),T(NH,1)	FORCI	5
2	,AMA(MB,MB)	FORCI	6
C		FORCI	7
	N=0	FORCI	8
C	NC=SMOOTHING IN SUBROUTINE SMOOTH. (SPLN1)	FORCI	9
	D=XX(2)-XX(1)	FORCI	10
	MZ=KSFY(MZ)	FORCI	11
	IF((KSFY(MZ).EQ.0.AND.NEWM.EQ.2) MZ=KSFY	FORCI	12
	DO 150 J=1,JMAX	FORCI	13
	MLTI=MLT(1,J)	FORCI	14
	NZ=MLT(2,J)-MLTI+2	FORCI	15
	NC=NZ+1	FORCI	16
	K1=1	FORCI	17
	K2=4	FORCI	18
	KC=0	FORCI	19
	T(1,1)=XTE(J)	FORCI	20
40	KK=MLTI	FORCI	21
	T(1,2)=XLE(J)	FORCI	22
	T(1,K2)=0.0	FORCI	23
	KA=	FORCI	24
	KB=NZ	FORCI	25
	IF(K1.EQ.3.OR.NZ.EQ.2) GO TO 45	FORCI	26
	IF(KC.EQ.2) GO TO 45	FORCI	27
C	EXCLUDE LEADING EDGE POINT IF IT IS TOO CLOSE TO FIRST BOX	FORCI	28
	IF (YA(KK)-XLE(J).GT.0.2*D) GO TO 45	FORCI	29
	KA=1	FORCI	30
	K5=NZ-1	FORCI	31
	KC=1	FORCI	32
45	CONTINUE	FORCI	33
	DO 50 K=KA,K5	FORCI	34
	T(K,2)=YA(KK)	FORCI	35
	IF(K1.EQ.3) GO TO 50	FORCI	36
	T(K,K2)=S(K1,J,KK)	FORCI	37
50	KK=KK+1	FORCI	38
	IF(KC.EQ.2) GO TO 51	FORCI	39
	IF(K1.EQ.3) GO TO 60	FORCI	40
	CALL SPLN1(1,T(1,1),T(1,8),T(1,9),KB,T(1,2),T(1,K2),T(1,10),N)	FORCI	41
	T(NZ,K2)=T(1,8)	FORCI	42
	IF(KC.NE.1) GO TO 51	FORCI	43
	KC=2	FORCI	44
	GO TO 40	FORCI	45
51	CONTINUE	FORCI	46
C		FORCI	47
	IF(NZ.GT.2) GO TO 52	FORCI	48
C	ADJUSTMENT FOR TRAILING EDGE VELOCITY POTENTIAL	FORCI	49
	KK=MLT(2,J)	FORCI	50
	DUM=SQRT((XTE(J)-XLE(J))/(XX(KK)-XLE(J)))	FORCI	51
	T(NZ,K2)=DUM*T(NZ,K2)	FORCI	52
52	CONTINUE	FORCI	53
C		FORCI	54
	KC=0	FORCI	55
	IF(K1.EQ.2) GO TO 55	FORCI	56
	K1=2	FORCI	57
	K2=6	FORCI	58
	GO TO 40	FORCI	59
55	K1=3	FORCI	60
	GO TO 40	FORCI	61
60	T(NZ,2)=XTE(J)	FORCI	62
	IF (ABS(XLE(J)-XX(MLTI)).GT.1.E-05) GO TO 67	FORCI	63
C	LEADING EDGE AND FIRST BOX COINCIDE	FORCI	64
	NZ=NZ-1	FORCI	65
	NC=NC-1	FORCI	66
	DO 65 K=2,NC	FORCI	67
	T(K,2)=T(K+1,2)	FORCI	68
	T(K,4)=T(K+1,4)	FORCI	69
	T(K,6)=T(K+1,6)	FORCI	70

IF (NEW,1,1) CC TO 15	FORCI	71
T(K,12)=T(K+1,12)	FORCI	72
65 CONTINUE	FORCI	73
67 CONTINUE	FORCI	74
C	FORCI	75
C INTERPOLATE DEFLECTION AT BOX CENTER, LEADING AND TRAILING EDGES	FORCI	76
CALL SUREZ(YY(J),T(1,1),1,NQ,2,T(1,13),T(1,14),DUM,SFDX(1,M2)	FORCI	77
5 SFCY(1,M2),SFDX(1,M2),+Z,2)	FORCI	78
C PERFORM SPANWISE INTEGRATION	FORCI	79
DO 75 K=1,N	FORCI	80
T(K,12)=T(K,14)*T(K,4)+CK*T(K,13)*T(K,6)	FORCI	81
75 T(K,16)=T(K,14)*T(K,6)-CK*T(K,13)*T(K,4)	FORCI	82
CALL INTGL(T(1,1),T(1,15),T(1,17),T(1,18),T(1,19),2,NQ,NZ,NM)	FORCI	83
DUM=T(NQ,13)	FORCI	84
T(J,17)=T(NQ,4)*DUM-T(J,17)	FORCI	85
T(J,18)=T(NQ,6)*DUM-T(J,18)	FORCI	86
150 CONTINUE	FORCI	87
C PERFORM SPANWISE INTEGRATION	FORCI	88
NQ=JMAK+2	FORCI	89
K1=17	FORCI	90
K2=15	FORCI	91
T(1,1)=0.0	FORCI	92
165 DO 170 K=1,JMAK	FORCI	93
170 T(K,2)=YY(K)	FORCI	94
NZ=JMAK	FORCI	95
CALL SFLN1(1,T(1,1),T(1,5),T(1,9),NZ,T(1,2),T(1,K1),T(1,10),N)	FORCI	96
T(NQ,K2)=0.0	FORCI	97
T(1,K2)=T(1,6)	FORCI	98
DO 175 K=1,JMAK	FORCI	99
175 T(K+1,K2)=T(K,K1)	FORCI	100
IF (K1.EQ.18) GO TO 180	FORCI	101
K1=18	FORCI	102
K2=16	FORCI	103
GO TO 165	FORCI	104
180 CONTINUE	FORCI	105
NZ=JMAK+1	FORCI	106
T(1,2)=0.0	FORCI	107
T(NQ,2)=YMAX	FORCI	108
DO 185 K=1,JMAK	FORCI	109
185 T(K+1,2)=YY(K)	FORCI	110
C WING TIP CORRECTION	FORCI	111
K1=0	FORCI	112
XD=T(NQ,2)-T(NZ-1,2)	FORCI	113
PR=T(NZ-1,15)	FORCI	114
PI=T(NZ-1,16)	FORCI	115
IF (T(NZ,2)+.5*D.LT.T(NQ,2)) GO TO 195	FORCI	116
190 DUM=SQRT((T(NQ,2)-T(NZ,2))/XD)	FORCI	117
T(NZ,15)=DUM*PR	FORCI	118
T(NZ,16)=DUM*PI	FORCI	119
IF (K1.EQ.1) GO TO 196	FORCI	120
195 IF (YY(JMAK)+1.05*D.GE.YMAX) GO TO 196	FORCI	121
K1=1	FORCI	122
NZ=NZ+1	FORCI	123
NQ=NQ+1	FORCI	124
T(NQ,2)=T(NQ-1,2)	FORCI	125
T(NZ,2)=T(NZ-1,2)+D	FORCI	126
T(NQ,15)=T(NQ-1,15)	FORCI	127
T(NQ,16)=T(NQ-1,16)	FORCI	128
GO TO 190	FORCI	129
196 CONTINUE	FORCI	130
IF (ABS(T(NQ,2)-T(NZ,2)).GE.1.E-05) GO TO 197	FORCI	131
NQ=NQ-1	FORCI	132
NZ=NZ-1	FORCI	133
T(NQ,2)=T(NQ+1,2)	FORCI	134
T(NQ,15)=T(NQ+1,15)	FORCI	135
T(NQ,16)=T(NQ+1,16)	FORCI	136
197 CONTINUE	FORCI	137
C	FORCI	138
CALL INTGL(T(1,2),T(1,15),T(1,17),T(2,1),T(1,19),2,NQ,NZ,NM)	FORCI	139
RETURN	FORCI	140
END	FORCI	141

	SUBROUTINE MRED(DA,T,NM,NB,KSFH,SFMH,SFMY,SFMH,IN,IPRINT)	MRED	2
C	SPLINE-SURFACE FIT OF MACH NUMBER	MRED	3
	DIMENSION DA(1),I(NM,1),SFMH(1),SFMY(1),SFMH(1)	MRED	4
	CONST=0.28571429	MRED	5
	KSFH=DA(97)	MRED	6
	IF(KSFH) 80,50,10	MRED	7
C		MRED	8
C	FITTING OF GIVEN PRESSURE/MACH TO A SPLINE-SURFACE	MRED	9
C		MRED	10
	10 IF(NB-KSFH) 80,15,15	MRED	11
	15 CONTINUE	MRED	12
	KP=701	MRED	13
	DO 30 IP=1,KSFH	MRED	14
	SFMH(IP)=DA(KP)/DA(24)	MRED	15
	SFMY(IP)=DA(KP+1)/DA(24)	MRED	16
	SFMH(IP)=DA(KP+2)	MRED	17
C	DA(96)=1, INPUT DATA ARE PRESSURE COEFFICIENT	MRED	18
C	DA(96)=2, INPUT DATA ARE LOCAL MACH NUMBER	MRED	19
	IF(DA(96)-1.0) 75,20,25	MRED	20
C	CONVERT PRESSURE COEFFICIENT INTO LOCAL MACH NUMBER	MRED	21
	20 SFMH(IP)=SQRT(5*(1.2/((1.+0.7*SFMH(IP))*CONST)-1.))	MRED	22
	25 CONTINUE	MRED	23
	30 KP=KP+3	MRED	24
C	SPLINE-SURFACE FITTING OF DATA	MRED	25
	40 CONTINUE	MRED	26
	IF(IPRINT.NE.C) WRITE(IN,100)	MRED	27
	CALL SURF1(NM,KSFH,T,SFMH,SFMY,SFMH,IPRINT)	MRED	28
	RETURN	MRED	29
C		MRED	30
C	PRESENTLY INPUT OF PRESSURE COEFFICIENT IN	MRED	31
C	A POLYNOMIAL FORM IS NOT ALLOWED	MRED	32
C	THE FOLLOWING IS FOR MACH INPUT AS A POLYNOMIAL $M=A0+A1*X$	MRED	33
	50 CONTINUE	MRED	34
	KSFH=0	MRED	35
	SFMH(1)=JA(71)	MRED	36
	SFMH(2)=DA(72)*DA(24)	MRED	37
	SFMH(3)=0.0	MRED	38
	GO TO 40	MRED	39
C		MRED	40
	75 IPR=96	MRED	41
	GO TO 85	MRED	42
	80 IPR=97	MRED	43
	85 WRITE	MRED	44
	C (IN,110)IPR	MRED	45
	STOP	MRED	46
	100 FORMAT(1H0,10X,73HCOMPUTED MACH(X,Y) = A0+A1*X+A2*Y+ SUM OF H(I)*	MRED	47
	S(R(I)**2)*(ALCG(R(I)**2)))	MRED	48
	110 FORMAT(1H0,10X,14HMRED--BAD DATA,15)	MRED	49
	END	MRED	50

	SUBROUTINE INTGL(X,Y,VR,VI,S,M,NQ,NZ,NM)	INTGL	2
C	INTEGRATION BASED ON SPLINE FUNCTION	INTGL	3
	DIMENSION X(1),Y(NM,2),S(NM,1)	INTGL	4
C	DEFINE L(I)	INTGL	5
	DO 20 I=2,NQ	INTGL	6
20	S(I,1)=X(I)-X(I-1)	INTGL	7
	IF (NQ.EQ.2) GO TO 50	INTGL	8
C	DEFINE TRI-DIAGONAL COEFFICIENT MATRIX	INTGL	9
	DC 25 I=2,NZ	INTGL	10
	S(I,2)= S(I,1)/6.0	INTGL	11
	S(I,3)=(S(I,1)+S(I+1,1))/3.0	INTGL	12
25	S(I,4)= S(I+1,1)/6.0	INTGL	13
	S(I,2)=0.0	INTGL	14
	S(NZ,4)=0.0	INTGL	15
C	DEFINE RIGHT-HAND-SIDE COLUMN MATRIX	INTGL	16
	K=1	INTGL	17
35	DO 40 I=2,NQ	INTGL	18
40	S(I,5)=(Y(I,K)-Y(I-1,K))/S(I,1)	INTGL	19
	DC 45 I=2,NZ	INTGL	20
45	S(I,6)= S(I+1,5)-S(I,5)	INTGL	21
C	SOLVE FOR COEFFICIENTS OF SPLINE FUNCTION M(J)	INTGL	22
	CALL TRIDI(2,NZ,S(I,2),S(I,3),S(I,4),S(I,6),S(I,7),S(I,8),S(I,9))	INTGL	23
50	CONTINUE	INTGL	24
	S(I,7)=0.0	INTGL	25
	S(NQ,7)=0.0	INTGL	26
	VI=0.0	INTGL	27
	DO 60 I=2,NQ	INTGL	28
	VI=VI+0.5*S(I,1)	INTGL	29
	S	INTGL	30
60	CONTINUE	INTGL	31
	IF(K.EQ.2) RETURN	INTGL	32
	VR=VI	INTGL	33
	IF(N.EQ.1) RETURN	INTGL	34
	K=2	INTGL	35
	GO TO 35	INTGL	36
	END	INTGL	37

	SUBROUTINE TRIDI(K1,K3,A,B,C,D,V,E,F)	TRIDI	2
	DIMENSION A(1),B(1),C(1),D(1),V(1),E(1),F(1)	TRIDI	3
	IF (K3.NE.K1) GO TO 5	TRIDI	4
	V(K3)=C(K3)/B(K3)	TRIDI	5
	RETURN	TRIDI	6
5	CONTINUE	TRIDI	7
	E(K1)=B(K1)	TRIDI	8
	F(K1)=D(K1)/E(K1)	TRIDI	9
	K2=K1+1	TRIDI	10
	CC 1) I=K2,K3	TRIDI	11
	L(I)=B(I)-A(I)*C(I-1)/E(I-1)	TRIDI	12
10	F(I)=(D(I)-A(I)*F(I-1))/E(I)	TRIDI	13
	V(K3)=F(K3)	TRIDI	14
	K2=K3-K1	TRIDI	15
	CC 2) J=1,K2	TRIDI	16
	I=K3-J	TRIDI	17
20	V(I)=F(I)-C(I)*V(I+1)/E(I)	TRIDI	18
	RETURN	TRIDI	19
	END	TRIDI	20

	SUBROUTINE SPLN1(N1,X,Y,DY,N,XX,YY,DYY,NSMOS)	SPLN1	2
C	X,Y,DY=INTERPOLATION INDEPENDENT, DEPENDENT VARIABLES, AND DY/DX	SPLN1	3
C	N1 =NO. OF INTERPOLATION POINTS	SPLN1	4
C	XX,YY =INPUT INDEPENDENT AND DEPENDENT VARIABLES	SPLN1	5
C	N =NO. OF INPUT POINTS	SPLN1	6
C	DYY =D(YY)/C(XX) FOR INPUT DATA	SPLN1	7
C	NSMOS =CONTROLS OF SMOOTHING AND PRE-INTERPOLATION	SPLN1	8
	1P,NSION X(1),Y(1),DY(1),XX(1),YY(1),DYY(1)	SPLN1	9
	N2=N	SPLN1	10
	IPRE=NSMOS/10	SPLN1	11
	NSML=NSMOS-IPRE*10	SPLN1	12
	IF(IPRE.EC.0) GO TO 30	SPLN1	13
C	STORE INPUT DATA FOR PRE-INTERPOLATION	SPLN1	14
	DO 25 J=1,N	SPLN1	15
	Y(J)=XX(J)	SPLN1	16
20	DY(J)=YY(J)	SPLN1	17
	CALL SPISET(N,XX,YY,DYY,0.0,0)	SPLN1	18
C	PRE-INTERPOLATION	SPLN1	19
	DO 25 I=2,N	SPLN1	20
	I1=I-1	SPLN1	21
	I2=I+1	SPLN1	22
	XX(I2)=Y(I)	SPLN1	23
	YY(I2)=DY(I)	SPLN1	24
	XX(I1)=0.5*(Y(I-1)+Y(I))	SPLN1	25
25	CALL SPLN2(XX(I1),1,N,Y,DY,DYY,YY(I1),DUM,1)	SPLN1	26
	N2=2*N-1	SPLN1	27
30	CONTINUE	SPLN1	28
	IF(NSPS.EC.0) GO TO 40	SPLN1	29
C	SMOOTH INPUT DATA XX, YY	SPLN1	30
	CALL SMOOTH(N2,XX,YY,Y,NSPJ)	SPLN1	31
C	INTERPOLATE Y AT X FROM XX, YY, DYY AND CALCULATE DY=D(Y)/D(X)	SPLN1	32
40	CALL SPISET(N2,XX,YY,DYY,0.0,0)	SPLN1	33
	CALL SPLN2(X,N1,N2,XX,YY,DYY,Y,DY,2)	SPLN1	34
	RETURN	SPLN1	35
	END	SPLN1	36

	SUBROUTINE SPLN2(XP,NP,N,X,Y,D,SPF,SPD,K)	SPLN2	2
	DIMENSION X(1),Y(1),D(1),XP(1),SPF(1),SPD(1)	SPLN2	3
C	EVALUATES A NATURAL CUBIC SPLINE AND ITS FIRST DERIVATIVE USING	SPLN2	4
C	SLOPE ARRAY D CALCULATED BY SPISET AND USING THE INPUT DATA	SPLN2	5
C	ARRAYS X AND Y	SPLN2	6
	DO 10 J=1,NP	SPLN2	7
	IF(XP(J).LT.X(1).OR.N.EQ.1) GO TO 6	SPLN2	8
	DO 2 I=2,N	SPLN2	9
	IF(XP(J).LT.X(I)) GO TO 4	SPLN2	10
2	CONTINUE	SPLN2	11
	SPF(J)=Y(N)+D(N)*(XP(J)-X(N))	SPLN2	12
	IF(K.EQ.1) GO TO 10	SPLN2	13
	SPD(J)=D(N)	SPLN2	14
	GO TO 10	SPLN2	15
4	C1=1./(X(I)-X(I-1))	SPLN2	16
	C2=X(I)-XP(J)	SPLN2	17
	C3=XP(J)-X(I-1)	SPLN2	18
	C4=C2*C1	SPLN2	19
	C5=C3*C1	SPLN2	20
	SPF(J)=C5*C5*((1.+2.*C4)*Y(I)-C2*D(I))	SPLN2	21
	+C4*C4*((1.+2.*C5)*Y(I-1)+C3*D(I-1))	SPLN2	22
	IF(K.EQ.1) GO TO 10	SPLN2	23
	C6=2.*C2-C3	SPLN2	24
	C7=2.*C3-C2	SPLN2	25
	SPD(J)=C1*C1*(C3*(2.*(1.+C1*C6)*Y(I)-C6*D(I))	SPLN2	26
	-C2*(2.*(1.+C1*C7)*Y(I-1)+C7*D(I-1)))	SPLN2	27
	GO TO 10	SPLN2	28
6	SPF(J)=Y(1)-D(1)*(X(I)-XP(J))	SPLN2	29
	IF(K.EQ.1) GO TO 10	SPLN2	30
	SPD(J)=D(1)	SPLN2	31
10	CONTINUE	SPLN2	32
	RETURN	SPLN2	33
	END	SPLN2	34



SUBROUTINE SPISET(N,X,Y,C,RMS,IFRMS)	SPISET	2
DIMENSION X(1),Y(1),D(1)	SPISET	3
DATA JMAX,FACT,PL,PI,IOBDE,BO7,1.57179677/	SPISET	4
D(1)=0.	SPISET	5
IF(N.EQ.1) RETURN	SPISET	6
D(N)=0.	SPISET	7
AN=N	SPISET	8
DO 1 I=2,N	SPISET	9
I=N+2-I	SPISET	10
X(I)=X(I)-X(I-1)	SPISET	11
1 Y(I)=(Y(I)-Y(I-1))/X(I)	SPISET	12
IF(N.EQ.2) GO TO 5	SPISET	13
DO 2 I=3,N	SPISET	14
I=N+3-I	SPISET	15
D(I-1)=2.*(Y(I)-Y(I-1))/(X(I)+X(I-1))	SPISET	16
Y(I)=1.5*D(I-1)	SPISET	17
2 X(I)=0.5*(X(I-1)/(X(I)+X(I-1)))	SPISET	18
DO 3 J=1,JMAX	SPISET	19
DO 3 I=3,N	SPISET	20
3 D(I-I)=X*(Y(I)-Y(I-1))+D(I-2)-(0.5-X(I))*D(I)-((N-1.0)*D(I-1))	SPISET	21
DO 4 I=3,N	SPISET	22
X(I)=X(I-1)*(C./X(I)-X.)	SPISET	23
4 Y(I)=1.33333333*Y(I)*(X(I)+X(I-1))+Y(I-1)	SPISET	24
5 SAVE=Y(I)-X*(X(I)+X(I-1))+Y(I-1)	SPISET	25
DO 6 I=3,N	SPISET	26
C=X(X(I))	SPISET	27
S1=Y(I)-C*(X(I)+X(I-1))+D(I)	SPISET	28
S2=Y(I)+C*(X(I)+X(I-1))+D(I)	SPISET	29
D(I-1)=0.5*(SAVE+S1)	SPISET	30
6 SAVE=S2	SPISET	31
D(N)=SAVE	SPISET	32
IF(IFRMS.NE.1) GO TO 5	SPISET	33
RMS=0.	SPISET	34
IF(N.EQ.2) GO TO 7	SPISET	35
DO 7 I=3,N	SPISET	36
C=2.*D(I-1)	SPISET	37
7 RMS=RMS+((C+D(I-2)-3.*Y(I-1))/X(I-1)+(D(I)+C-3.*Y(I))/X(I))**2	SPISET	38
RMS=2.*SQRT(RMS/AN)	SPISET	39
8 DO 9 I=2,N	SPISET	40
Y(I)=Y(I)*X(I)+Y(I-1)	SPISET	41
9 X(I)=X(I)+X(I-1)	SPISET	42
RETURN	SPISET	43
END	SPISET	44

	SUBROUTINE SMOOTH (N,X,Y,T,NSMDS)	SMOOTH	2
C	THE Y ARRAY IS SMOOTHED BY A LOCAL FIVE POINT LEAST SQUARES	SMOOTH	3
C	CUBIC WEIGHTED BY M	SMOOTH	4
	DIMENSION X(1),Y(1),T(1)	SMOOTH	5
	IF(N.LT.5) RETURN	SMOOTH	6
	DO 10 NS=1,NSMDS	SMOOTH	7
	T(1)=NS	SMOOTH	8
	AN=N	SMOOTH	9
	S=(T(1)+(X(N)-X(1))/AN)**2	SMOOTH	10
	DO 4 L=1,M	SMOOTH	11
	K=MINO(M-4,MAXO(1,L-2))	SMOOTH	12
	K4=K+4	SMOOTH	13
	DO 1 I=1,20	SMOOTH	14
1	T(I)=0.	SMOOTH	15
	DO 3 M=K,K4	SMOOTH	16
	W=1./(S+(X(L)-X(M))**2)	SMOOTH	17
	R=1.0	SMOOTH	18
	DO 3 I=1,4	SMOOTH	19
	I4=I-4	SMOOTH	20
	RR=1.0	SMOOTH	21
	DO 2 J=1,4	SMOOTH	22
	J4=4*J+I4	SMOOTH	23
	T(J4)=T(J4)+R*RR*W	SMOOTH	24
2	RR=RR*X(M)	SMOOTH	25
	T(I+16)=T(I+16)+R*Y(M)*W	SMOOTH	26
3	R=R*X(M)	SMOOTH	27
	CALL CHLSKY(T,4,T(17),1,4)	SMOOTH	28
	M=L-((L-1)/5)*5	SMOOTH	29
	IF(L.GT.5) Y(L-5)=T(M+20)	SMOOTH	30
	T(M+20)=0.	SMOOTH	31
	R=1.0	SMOOTH	32
	DO 4 J=1,4	SMOOTH	33
	T(M+20)=T(M+20)+R*T(J+16)	SMOOTH	34
4	R=R*X(L)	SMOOTH	35
	I4=N-5	SMOOTH	36
	DO 5 L=1,5	SMOOTH	37
	ML=M+L-((M+L-1)/5)*5	SMOOTH	38
	J4=I4+L	SMOOTH	39
5	Y(J4)=T(ML+20)	SMOOTH	40
10	CONTINUE	SMOOTH	41
	RETURN	SMOOTH	42
	END	SMOOTH	43

	SUBROUTINE CHLSKY(A,N,B,M,NX)	CHLSKY	2
	DIMENSION A(NX,1),B(NX,1)	CHLSKY	3
C	CHOLESKY DECOMPOSITION IS USED TO SOLVE THE MATRIX EQUATION AX=B	CHLSKY	4
C	WHERE THE COEFFICIENT MATRIX, A, IS SYMMETRIC. ON OUTPUT X IS	CHLSKY	5
C	STORED IN B	CHLSKY	6
	IF(N.EQ.1) GO TO 6	CHLSKY	7
	DO 2 I=2,N	CHLSKY	8
	I1=I-1	CHLSKY	9
	DO 2 J=I,N	CHLSKY	10
	DO 2 L=1,I1	CHLSKY	11
	2 A(I,J)=A(I,J)-A(L,I)*A(L,J)/A(L,L)	CHLSKY	12
	DO 5 K=1,M	CHLSKY	13
	DO 3 I=2,N	CHLSKY	14
	I1=I-1	CHLSKY	15
	DO 3 L=1,I1	CHLSKY	16
	3 B(I,K)=B(I,K)-A(L,I)*B(L,K)/A(L,L)	CHLSKY	17
	DO 4 I=2,N	CHLSKY	18
	I1=I-1	CHLSKY	19
	DO 4 L=1,I1	CHLSKY	20
	NI=N-I1	CHLSKY	21
	NL=N+1-L	CHLSKY	22
	4 B(NI,K)=B(NI,K)-A(NI,NL)*B(NL,K)/A(NL,NL)	CHLSKY	23
	DO 5 I=1,N	CHLSKY	24
	5 B(I,K)=B(I,K)/A(I,I)	CHLSKY	25
	RETURN	CHLSKY	26
	6 A(I,I)=1./A(I,I)	CHLSKY	27
	DO 7 L=1,M	CHLSKY	28
	7 B(I,L)=A(I,I)*B(I,L)	CHLSKY	29
	RETURN	CHLSKY	30
	END	CHLSKY	31

	SUBROUTINE SUPFLINM,N,T,ABX,ABY,ABH,IRITE)	SURF1	2
C	FIT DATA (N POINTS) BROUGHT THROUGH ABX,ABY,ABH	SURF1	3
C	1 -- TEMPORARY ARRAY FOR SPLINE-SURFACE FITTING	SURF1	4
C	2 -- POINTS ARRAYS FOR INPUT POINTS TO BE FITTED	SURF1	5
C	ABX(1) - INDEPENDENT VARIABLE X	SURF1	6
C	ABY(1) - INDEPENDENT VARIABLE Y	SURF1	7
C	ABH(1) - COMES IN AS DEPENDENT VARIABLE OF X AND Y	SURF1	8
C	GOES OUT AS COEFFICIENTS OF SPLINE-SURFACE	SURF1	9
C	IF (N.EQ.0) GO TO 13	SURF1	10
C	13 --	SURF1	11
C	14 --	SURF1	12
C	NP1=N +1	SURF1	13
C	NP2=NP1+1	SURF1	14
C	NP3=NP2+1	SURF1	15
C	NP4=NP3+1	SURF1	16
C	IF (N.EQ.0) GO TO 13	SURF1	17
C	DO 2 I=1,N	SURF1	18
C	T(I,NP2)=ABX(I)	SURF1	19
C	T(I,NP3)=ABY(I)	SURF1	20
C	2 T(I,NP4)=ABH(I)	SURF1	21
C	DO 4 I=1,N	SURF1	22
C	T(I,I)=0.	SURF1	23
C	T(I,NP1)=1.0	SURF1	24
C	T(NP1,I)=1.0	SURF1	25
C	T(NP2,I)=T(I,NP2)	SURF1	26
C	4 T(NP3,I)=T(I,NP3)	SURF1	27
C	NM1=N-1	SURF1	28
C	DO 5 I=1,NM1	SURF1	29
C	IP1=I+1	SURF1	30
C	DO 6 J=IP1,N	SURF1	31
C	XX=T(I,NP2)-T(J,NP2)	SURF1	32
C	YY=T(I,NP3)-T(J,NP3)	SURF1	33
C	H=XX*XX+YY*YY	SURF1	34
C	T(I,J)=H*ALOG(H)	SURF1	35
C	6 T(J,I)=T(I,J)	SURF1	36
C	DO 8 I=1,3	SURF1	37
C	IPN=I+N	SURF1	38
C	DO 8 J=1,4	SURF1	39
C	JPN=J+N	SURF1	40
C	8 T(IPN,JPN)=0.	SURF1	41
C	K=MSIMR(NM,NP3,1,T(1,1),T(1,NP4))	SURF1	42
C	IF (K.EQ.1) GO TO 9	SURF1	43
C	WRITE (14,220)	SURF1	44
C	STOP	SURF1	45
C	9 CONTINUE	SURF1	46
C	STOPE (N+3) - COEFFICIENT IN ARRAY ABH	SURF1	47
C	DO 12 I=1,NP3	SURF1	48
C	12 ABH(I)=T(I,NP4)	SURF1	49
C	13 IF (IRITE) 18,18,14	SURF1	50
C	14 WRITE (14,200)	SURF1	51
C	WRITE (14,120) (ABH(I),I=NP1,NP3)	SURF1	52
C	IF (N.EQ.0) GO TO 18	SURF1	53
C	DO 16 I=1,N	SURF1	54
C	16 WRITE (14,110) I,ABH(I),ABX(I),ABY(I)	SURF1	55
C	18 CONTINUE	SURF1	56
C	RETURN	SURF1	57
C	110 FORMAT(10X,I5,1P3E14.7)	SURF1	58
C	120 FORMAT(62X,1P3E14.7)	SURF1	59
C	200 FORMAT('140,20X,57HHERE X(1)+Z=(X-X(1))*Z+(Y-Y(1))*Z/ 11H0,10X,54HIN DIMENSIONLESS COORDINATES - DISTANCE/CHORD LENGTH)/ 21H0,13X,1P1,3X,4HH(I),10X,4Hx(I),10X,4HY(I), 315X,2HAG,12X,2HA1,12X,2HA2/)	SURF1	60
C	220 FORMAT(3X,2CHEID NOT CONVERGE IN SURF1//)	SURF1	61
C	END	SURF1	62
C		SURF1	63
C		SURF1	64
C		SURF1	65

	SUBROUTINE SURF2(Z1,Z2,J1,J2,MXY,VALU,VLUX,VLUY,XI,YI,HI,N,H)	SURF2	2
C	COMPUTE VALUE OF SPLINE-SURFACE FITTED DATA AT A POINT (X,Y)	SURF2	3
C	Z1,Z2- COORDINATES OF THE POINT WHERE THE FITTED VALUE IS SOUGHT	SURF2	4
C	VALU - FITTED VALUE SOUGHT	SURF2	5
C	VLUX - GRADIENT OF FITTED VALUE IN X	SURF2	6
C	VLUY - GRADIENT OF FITTED VALUE IN Y	SURF2	7
C	XI,YI,HI - ARRAYS FOR KNOWN PROPERTIES IN SPLINE-SURFACE FORM	SURF2	8
C	N -- NUMBER OF POINTS IN XI, YI ARRAYS	SURF2	9
C	MXY=0 X=Z1(J), Y=Z2(J) WHERE J=J1,J2	SURF2	10
C	MXY=1 X=Z1(1), Y=Z2(J) WHERE J=J1,J2	SURF2	11
C	MXY=2 Y=Z1(1), X=Z2(J) WHERE J=J1,J2	SURF2	12
	DIMENSION XI(1),YI(1),HI(1)	SURF2	13
	DIMENSION Z1(1),Z2(1),VALU(1),VLUX(1),VLUY(1)	SURF2	14
	NP1=N +1	SURF2	15
	NP2=NP1+1	SURF2	16
	NP3=NP2+1	SURF2	17
	IF(MXY.EQ.1) X=Z1(1)	SURF2	18
	IF(MXY.EQ.2) Y=Z1(1)	SURF2	19
	DO 40 J=J1,J2	SURF2	20
	IF(MXY.EQ.1) Y=Z2(J)	SURF2	21
	IF(MXY.EQ.2) X=Z2(J)	SURF2	22
	IF(MXY.NE.0) GO TO 10	SURF2	23
	X=Z1(J)	SURF2	24
	Y=Z2(J)	SURF2	25
10	CONTINUE	SURF2	26
	IF (K-2) 13,12,11	SURF2	27
11	VLUY(J)=HI(NP3)	SURF2	28
12	VLUX(J)=HI(NP2)	SURF2	29
13	VALU(J)=HI(NP1)+HI(NP2)*X+HI(NP3)*Y	SURF2	30
	IF(N.EQ.0) GO TO 40	SURF2	31
	DO 30 I=1,N	SURF2	32
	TX=X-XI(I)	SURF2	33
	TY=Y-YI(I)	SURF2	34
	H = TX*TX+TY*TY	SURF2	35
	HA=0.	SURF2	36
	IF(H.GT.0.) HA=ALOG(H)	SURF2	37
	H8=2.*(1.+HA)*HI(I)	SURF2	38
	IF (K-2) 23,22,21	SURF2	39
21	VLUY(J)=VLUY(J)+H8*TY	SURF2	40
22	VLUX(J)=VLUX(J)+H8*TX	SURF2	41
23	VALU(J)=VALU(J)+HI(I)*H*HA	SURF2	42
30	CONTINUE	SURF2	43
40	CONTINUE	SURF2	44
	RETURN	SURF2	45
	END	SURF2	46

	FUNCTION CIN(X1,S)	CIN	2
C	SINE AND COSINE INTEGRAL SUBROUTINE	CIN	3
C		CIN	4
C	IF CALLED BY THE STATEMENT C=CIN(X,S)	CIN	5
C	C AND S ARE THE INTEGRALS OVER T FROM 1 TO INFINITY OF	CIN	6
C	COS(TX)/T AND SIN(TX)/T	CIN	7
C		CIN	8
	SG=1.0	CIN	9
	X=X1	CIN	10
	IF (X) 1,2,2	CIN	11
	1 SG=-SG	CIN	12
	X=-X	CIN	13
	2 X2=X*X	CIN	14
	IF (X-1.0) 3,3,4	CIN	15
C		CIN	16
C	FOR ABS(X) LESS THAN 1 A SERIES EXPANSION IS USED	CIN	17
C		CIN	18
	3 V=((X2/98.0-0.6)*.05*X2+1.0)*X2/18.0-1.0)*X+1.57079633	CIN	19
	U=((X2/45.0-1.0)*X2/24.0+1.0)*X2/4.0-.577215665-ALOG(X)	CIN	20
	GO TO 5	CIN	21
C		CIN	22
C	FOR ABS(X) GREATER THAN 1 APPROXIMATIONS OF HASTINGS ARE USED	CIN	23
C		CIN	24
	4 P=((X2+19.394119)*X2+47.411538)*X2+8.493336)/(((X2+21.361055)	CIN	25
	1 *X2+70.376496)*X2+30.038227)*X)	CIN	26
	Q=((X2+21.383724)*X2+49.719775)*X2+5.089504)/(((X2+27.177958)	CIN	27
	1 *X2+119.918932)*X2+76.707876)*X2)	CIN	28
	CO=COS (X)	CIN	29
	SI=SIN (X)	CIN	30
	U=Q*CO-P*SI	CIN	31
	V=P*CO+Q*SI	CIN	32
	5 S=V*SG	CIN	33
	CIN=U	CIN	34
	RETURN	CIN	35
	END	CIN	36

FUNCTION MSIMER(M,N,L,A,B)	MSIMER	2
DIMENSION A(M,1),B(M,1)	MSIMER	3
DO 30 I = 1,N	MSIMER	4
C = 0.0	MSIMER	5
DO 10 J = 1,N	MSIMER	6
10 C = AMAX1(C,ABS(A(I,J)))	MSIMER	7
IF(C.EQ.0.0) GO TO 1000	MSIMER	8
DO 20 J = 1,N	MSIMER	9
20 A(I,J) = A(I,J)/C	MSIMER	10
DO 30 J = 1,L	MSIMER	11
30 B(I,J) = B(I,J)/C	MSIMER	12
IF(N.EQ.1) GO TO 205	MSIMER	13
NM = N - 1	MSIMER	14
DO 200 J = 1,NM	MSIMER	15
C = 0.0	MSIMER	16
K = 0	MSIMER	17
DO 40 I = J,N	MSIMER	18
D = ABS(A(I,J))	MSIMER	19
IF (C.GE.D) GO TO 40	MSIMER	20
K = I	MSIMER	21
C = D	MSIMER	22
40 CONTINUE	MSIMER	23
IF(K.EQ.0,OR,C.LT.1.E-7) GO TO 1000	MSIMER	24
IF(K.EQ.J) GO TO 70	MSIMER	25
DO 50 JJ = J,N	MSIMER	26
C = A(J,JJ)	MSIMER	27
A(J,JJ) = A(K,JJ)	MSIMER	28
50 A(K,JJ) = C	MSIMER	29
DO 60 JJ = 1,L	MSIMER	30
C = B(J,JJ)	MSIMER	31
B(J,JJ) = B(K,JJ)	MSIMER	32
60 B(K,JJ) = C	MSIMER	33
70 C = A(J,J)	MSIMER	34
JP = J + 1	MSIMER	35
DO 80 JJ = JP,N	MSIMER	36
80 A(J,JJ) = A(J,JJ)/C	MSIMER	37
90 DO 100 JJ = 1,L	MSIMER	38
100 B(J,JJ) = B(J,JJ)/C	MSIMER	39
DO 200 I = 1,N	MSIMER	40
IF(I.EQ.J) GO TO 200	MSIMER	41
C = A(I,J)	MSIMER	42
DO 110 JJ = JP,N	MSIMER	43
110 A(I,JJ) = A(I,JJ) - C*A(J,JJ)	MSIMER	44
DO 120 JJ = 1,L	MSIMER	45
120 B(I,JJ) = B(I,JJ) - C*B(J,JJ)	MSIMER	46
200 CONTINUE	MSIMER	47
205 C = A(N,N)	MSIMER	48
IF(ABS(C).LT.1.E-7) GO TO 1000	MSIMER	49
DO 210 J = 1,L	MSIMER	50
210 B(N,J) = B(N,J)/C	MSIMER	51
IF(N.EQ.1) GO TO 230	MSIMER	52
DO 220 I = 1,NM	MSIMER	53
C = A(I,N)	MSIMER	54
DO 220 JJ = 1,L	MSIMER	55
220 B(I,JJ) = B(I,JJ) - C*B(N,JJ)	MSIMER	56
230 MSIMER = 1	MSIMER	57
RETURN	MSIMER	58
1000 MSIMER = 2	MSIMER	59
RETURN	MSIMER	60
END	MSIMER	61

FUNCTION MSINEC(M,N,L,A,B)	MSINEC	1
DIMENSION A(M,1),B(N,1)	MSINEC	2
COMPLEX A,B,G	MSINEC	3
DO 30 I = 1,N	MSINEC	4
C = 0.0	MSINEC	5
DO 10 J = 1,M	MSINEC	6
10 C=AMAX1(C,ABS(REAL(A(I,J)))+ABS(AIMAG(A(I,J))))	MSINEC	7
IF(C.EQ.0.0) GO TO 1000	MSINEC	8
DO 20 J = 1,N	MSINEC	9
20 A(I,J) = A(I,J)/C	MSINEC	10
DO 30 J = 1,L	MSINEC	11
30 B(I,J) = B(I,J)/C	MSINEC	12
IF(N.EQ.1) GO TO 205	MSINEC	13
MM = M - 1	MSINEC	14
DO 200 J = 1,MM	MSINEC	15
C = 0.0	MSINEC	16
K = 0	MSINEC	17
DO 40 I = J,M	MSINEC	18
G=ABS(REAL(A(I,J)))+ABS(AIMAG(A(I,J)))	MSINEC	19
IF(C.GE.0) GO TO 40	MSINEC	20
K = I	MSINEC	21
C = 0	MSINEC	22
40 CONTINUE	MSINEC	23
IF(K.EQ.0.OR.C.LT.1.E-7) GO TO 1000	MSINEC	24
IF(K.EQ.J) GO TO 70	MSINEC	25
DO 50 JJ = J,M	MSINEC	26
G = A(J,JJ)	MSINEC	27
A(J,JJ) = A(K,JJ)	MSINEC	28
50 A(K,JJ) = G	MSINEC	29
DO 60 JJ = 1,L	MSINEC	30
G = B(J,JJ)	MSINEC	31
B(J,JJ) = B(K,JJ)	MSINEC	32
60 B(K,JJ) = G	MSINEC	33
70 G = 1.0/A(J,J)	MSINEC	34
JP = J + 1	MSINEC	35
DO 80 JJ = JP,N	MSINEC	36
80 A(J,JJ) = A(J,JJ)*G	MSINEC	37
90 DO 100 JJ = 1,L	MSINEC	38
100 B(J,JJ) = B(J,JJ)*G	MSINEC	39
DO 200 I = 1,M	MSINEC	40
IF(I.EQ.J) GO TO 200	MSINEC	41
G = A(I,J)	MSINEC	42
DO 110 JJ = JP,N	MSINEC	43
110 A(I,JJ) = A(I,JJ) - G*A(J,JJ)	MSINEC	44
DO 120 JJ = 1,L	MSINEC	45
120 B(I,JJ) = B(I,JJ) - G*B(J,JJ)	MSINEC	46
200 CONTINUE	MSINEC	47
205 G = A(N,N)	MSINEC	48
IF (ABS(REAL(G)) + ABS(AIMAG(G)).LT.1.E-7) GO TO 1000	MSINEC	49
DO 210 J = 1,L	MSINEC	50
210 E(N,J) = B(N,J)/G	MSINEC	51
IF(N.EQ.1) GO TO 230	MSINEC	52
DO 220 I = 1,M	MSINEC	53
DO 220 JJ = 1,L	MSINEC	54
220 B(I,JJ) = B(I,JJ) - A(I,N)*E(N,JJ)	MSINEC	55
230 MSINEC = 1	MSINEC	56
RETURN	MSINEC	57
1000 MSINEC = 2	MSINEC	58
RETURN	MSINEC	59
END	MSINEC	60
	MSINEC	61



1. Report No. NASA CR-158907		2. Government Accession No.		3. Recipient's Catalog No.	
4. Title and Subtitle SONIC-BOX METHOD EMPLOYING LOCAL MACH NUMBER FOR OSCILLATING WINGS WITH THICKNESS				5. Report Date September 1978	
				6. Performing Organization Code	
7. Author(s) S. Y. Ruo				8. Performing Organization Report No. LG78ER0226	
				10. Work Unit No.	
9. Performing Organization Name and Address Lockheed-Georgia Company 86 South Cobb Drive Marietta, Georgia 30063				11. Contract or Grant No. NAS1-13613	
				13. Type of Report and Period Covered Contractor Report	
12. Sponsoring Agency Name and Address National Aeronautics and Space Administration Washington, D. C. 20546				14. Sponsoring Agency Code	
15. Supplementary Notes Technical Monitor, Dr. E. Carson Yates, Jr., Aeroelasticity Branch, Structures and Dynamics Division, NASA-Langley Research Center, Hampton, Virginia 23365 Final Report					
16. Abstract A computer program has been developed to account approximately for the effects of finite wing thickness in the transonic potential flow over an oscillating wing of finite span. The program is based on the original sonic-box program for planar wing which has previously been extended to include the effects of the swept trailing edge and the thickness of the wing. Account for the non-uniform flow caused by finite thickness is made by application of the local linearization concept. The thickness effect, expressed in terms of the local Mach number, is included in the basic solution to replace the coordinate transformation method used in the earlier work. Calculations were made for a delta wing and a rectangular wing performing plunge and pitch oscillations, and the results were compared with those obtained from other methods. An input guide and a complete listing of the computer code are presented.					
17. Key Words (Suggested by Author(s)) Unsteady Transonic Flow Lifting Surface Theory Transonic Potential Flow Aircraft Aerodynamics			18. Distribution Statement  Unclassified - Unlimited STAR Category: 02		
19. Security Classif. (of this report) Unclassified	20. Security Classif. (of this page) Unclassified	21. No. of Pages 71	22. Price* \$5.25		